ファミコン大作戦

ファミコンソフト集

きみのファミコンは
ゲームがいっぱい

地球防衛軍 編著
司令 岩田 裕 谷岡康則

002

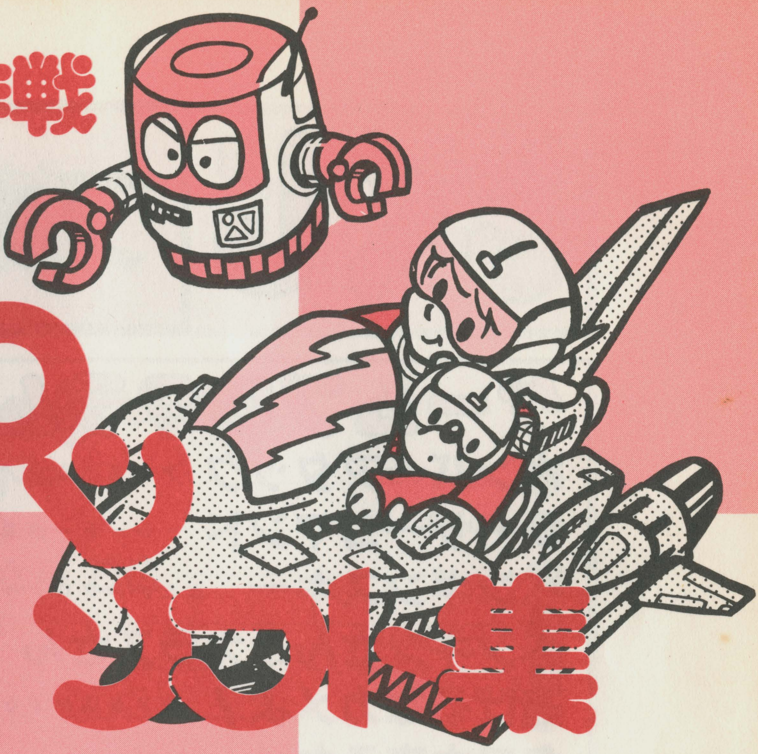


アクションゲームに
パズルゲーム

おもしろゲームいろいろ
16本だ！

誠文堂新光社

ファミコン大作戦



ファミコンソフト集

002

はじめに

ファミコンソフト集　００１は，どうでした？
「ファミリーベーシック」ホップ・ステップ・ジャンプは読んでもらえましたか？
さて，いよいよ
　　地球防衛軍が放つ
　　　第２弾　ファミコンソフト集　００２
の登場です。

1982バイトという，せまい空間につめこんだゲームの数々。００２には，パソコン用のゲームとしてはクラシックなものから最新のものまで，16本のゲームが入っています。

リアルタイム・アクションゲームが多かった００１集にくらべ，今回は，じっくり考える思考型タイプのゲームが多くなっています。

「あんな小さなメモリーじゃなにもできないよ。」と言っている友達に挑戦しましょう。

「コンピューターと言ってもオモチャだろう」と言う，お父さんにやってもらいましょう。

アイデアと，プログラミング・テクニックさえあれば，1982バイトは広大な宇宙です。

その空間が，何を君達に運んで行くか。

この本は僕達から君達への，ファミコンを通して送る第２のメッセージ集です。

「ファミリーベーシック」ホップ・ステップ・ジャンプを読んでベーシックと友達になった君，ベーシックなんか，とっくに知ってる君，僕達と一緒にゲーム・ワールドであそびましょう。

1985年　6月
地球防衛軍　スタッフ一同

もくじ ファミコン

ゲームプログラム集

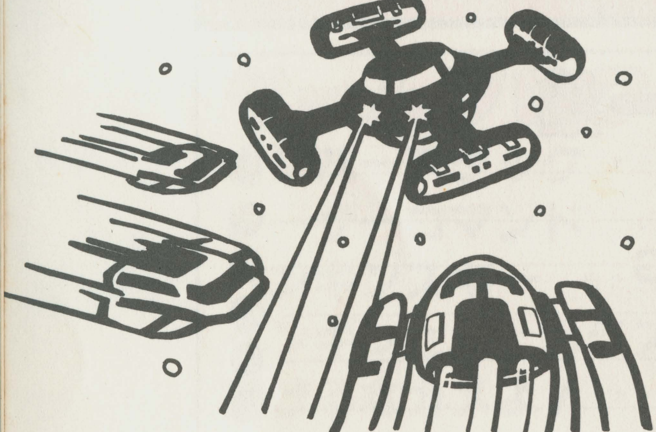
ソフト集 002

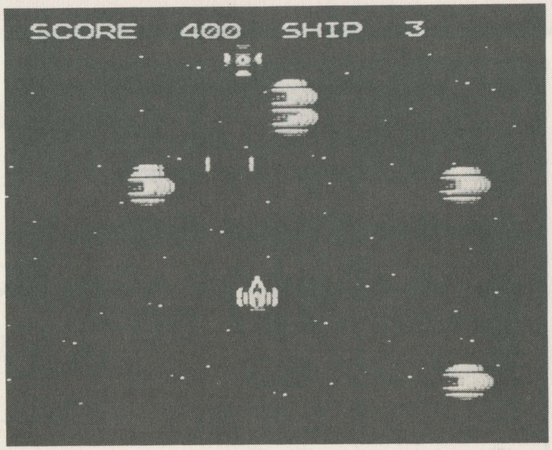
BG画面データー集

巻末資料

●制作・企画　地球防衛軍　岩田　裕・谷岡康則・鈴木康夫・真藤美紀子
簑藤　桂・北村　順・初瀬　籠・美崎靖之
●編集・企画　リポート社
●カバー・本文イラスト　高橋裕之（ファクトリー）

ボンバーアタック





敵の宇宙船は第１，第２の防衛線を突破して地球上に降りて来た。母船であるスピナーは護衛のスターキラーに守られながら，上空に浮かんでいる。ここで敵をやっつけなければ地上は敵の手に渡ってしまう。それは何としても止めねばならぬ。

地球上では残された５機のスターシップに，リモコンと爆弾を積んで，マザーシップに体当りをする計画をたてた。君がコマンダーを使って，敵のマザーシップを爆破してくれないか。

★遊び方

リスト通りにプログラムを打ち込んで，間違いがないとわかったら，ＢＧグラフィックのデータをキーボードから入力する。プログラムを RUNすると，ゲームが始まる。ゲームはコントローラーＩを使う。

画面に，Ａのボタンを押す様に指示が出るので，コントローラーのＡのボタンを押すと，画面が変わり，敵が動き出す。画面の下にある宇宙船が君のあやつる船，画面上を左から右に動いているのが，敵の護衛船，画面の上の方で止まっているのが，目標となる敵のマザーシップだ。護衛船の間を，うまく敵に当らないように操作して，マザーシップまで到達すれば良い。ただし，敵マザーシップの落とす爆戦に触れると，爆発してしまうので注意するこ

プログラムリスト

```
10 CGEN 2:CGSET 0,1:SPRITE ON:VIEW:SC=0:SH=5
15 DIMA(7)
20 FOR I=0 TO 1:DEF MOVE(I)=SPRITE(12,5,1,110,0,0):NEXT
30 FOR I=2 TO 7:DEF MOVE(I)=SPRITE(8,3,RND(3)+1,255,0,0):NEXT
40 DEF SPRITE0,(0,1,0,0,0)=CHR$(172)+CHR$(173)+CHR$(174)+CHR$(175):DEF SPRITE1,(1,0,1,0,0)=CHR$(176)+CHR$(177)+CHR$(178)+CHR$(179)
50 DEF SPRITE2,(0,1,0,0,0)=CHR$(180)+CHR$(181)+CHR$(182)+CHR$(183):DEF SPRITE3,(0,1,0,0,0)=CHR$(144)+CHR$(145)+CHR$(146)+CHR$(147)
60 X=120:Y=199:PLAY"T5Y2M1V5"
65 LOCATE 0,0:PRINT"      Aﾉ ﾎﾞﾀﾝ ｦ ｵｼﾃ ｸﾀﾞｻｲ     "
66 IF STRIG(0)<>8 THEN 66
67 LOCATE 0,0:PRINT"                            "
70 LOCATE 4,0:PRINT"SCORE ";SC:LOCATE 16,0:PRINT"SHIP ";SH
80 SPRITE3,120,32
90 FOR I=2 TO 7:GOSUB 500:NEXT
100 SPRITE 0,X,Y:XX=X:YY=Y:IF X<136 AND X>114 AND Y<56 THEN 800
110 IF STRIG(0)=8 THEN Y=Y-4
130 IF STICK(0)=1 THEN X=X+4
140 IF STICK(0)=2 THEN X=X-4
150 IF X<10 OR X>250 THEN X=XX
160 IF Y<34 THEN 710
170 I=0
```

と。移動は，コントローラーのジョイスティックの右および左，そしてボタンAで行なう。ボタンAを押している間だけ船は前に進み，左右へはジョイスティックを押している間だけ移動できる。この3つのボタンを組合わせて，前，ななめ前，左右の運動ができる。

敵は1機につき50点で，2000点で5機のスターシップがもらえる。5機ともやられてしまうと，ゲームオーバーとなり，画面が消えて，GAME　OVERと表示される。

ゲームを続けたい時は，キーボードのキーを押すと，再び最初に戻る。

★プログラムの説明

10行から60行が，初期設定です。10行のSHという変数が，スターシップの数ですから，もし点が取りにくい様な場合は，SHを適当に増してください。

65行から280行までが，メインプログラムです。

500行から520行が，マザーシップからの攻撃のパターンを決めているサブルーチンです。RND関数の数を変えると，爆弾が多くも少なくもなります。

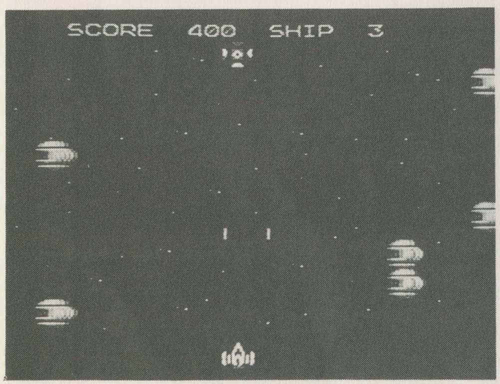
600行から620行のサブルーチンは，敵の護衛船の発生を行なうものです。

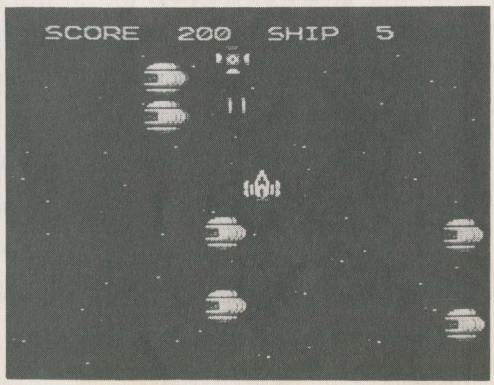
700行から820行で，爆発を行ないます。

900行から930行のサブルーチンは，ゲームオーバーになった時の処理を行なうものです。通常，BASICに戻れるようにするのならば，このルーチンの中で，抜け道を作ります。このプログラムの場合，ゲームをやめる時は，STOPキーを押すようになっています。

プログラムを改造する時のヒントとして少し書いておきます。

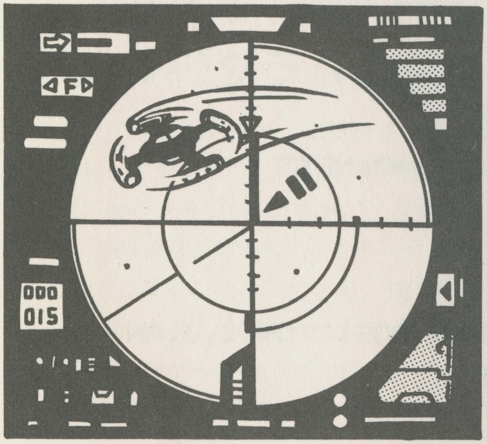
10行で，SCはスコアの入っている変数です。SHは味方の船の数です。**30行**のSPRITE命令の中のRND（3）＋1で，敵の護衛船の速さを決めていますので，この数を多くすると，敵はゆっくりと進む様になります。プログラム中でのX，Yは，自分の船の場所を示すものです。

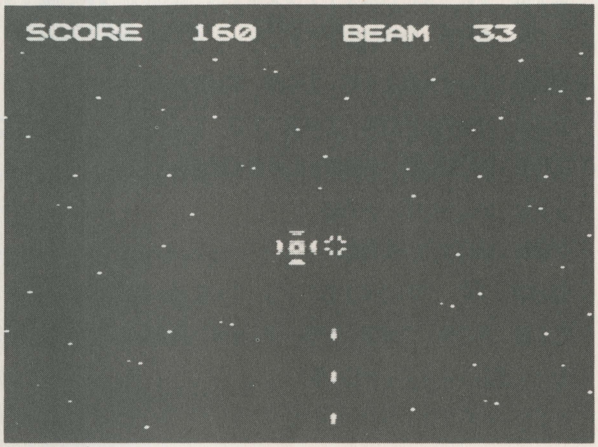




```
180 IF XPOS(I)>X-8AND XPOS(I)<X+16 ANDYPOS(I)>Y-8
ANDYPOS(I)<Y+16 THEN 700
190 I=I+1:IF I=2 THEN 210
200 GOTO 180
210 IF XPOS(I)>X-16 AND XPOS(I)<X+16 AND YPOS(I)>Y
-16 AND YPOS(I)<Y+16 THEN 700
220 I=I+1:IF I<8 THEN 210
230 N=0:FOR I=0 TO 7:A(I)=0:NEXT
250 GOSUB 500:N=0:FOR I=0 TO 1:A(I)=0:NEXT
270 GOSUB 600
280 GOTO 100
500 IF RND(10)>3 THEN RETURN
510 FOR I=2 TO 7:IF MOVE(I)=0 THEN POSITION I,0,RN
D(150)+40:MOVE I
520 NEXT:RETURN
600 IF RND(10)>5 THEN RETURN
610 FOR I=0 TO 1:IF MOVE(I)=0 THEN POSITION I,RND(
30)+110,32:MOVE I
620 NEXT:RETURN
700 ERA(I):SPRITE 1,X,Y:PLAY"O1C:O1G:O0C":SPRITE 2
,X,Y
710 SH=SH-1:IF SH=0 THEN 900
720 GOTO 20
800 SPRITE 1,X,Y:PLAY"O4C:O5E:O5G":SPRITE2,X,Y
810 SC=SC+50:IF SC=2000 THEN SH=SH+5
820 GOTO 20
900 SPRITE OFF:CGEN 2:LOCATE 8,10:PRINT"GAME OVER"
910 LOCATE 6,12:PRINT "ｷｰ ｦ ｵｼﾃ ｸﾀﾞｻｲ!"
920 K$=INKEY$:IF K$="" THEN 920
930 CLEAR:GOTO 10
```

レーザーコマンド





西暦201Ｘ年，ようやくラグランジェポイントに恒久的な天体観測および実験用のベースを築いた人類は，実用段階に入ったスペースシャトルを，その運送手段として，深遠なる宇宙に，その目を向けだした。

ある日，新星からの水素波をとらえるためにぎょしゃ座の方にアンテナを向けていたベースは，新しい電波源を発見した。その報告が地上に届いてから数週間後，ベースは，交戦中との連絡をした後，報告を断った。

地上の国との争いではないとわかって，各国は色めき立った。急いで，運行可能なスペースシャトルを高々度衛星軌道に集め，おそまつながら，その正体の見えざる敵に，レーザー誘導の旧式のミサイルで戦おうというのだ。

敵は大型の母船らしく，小型船用の兵器は積んでないらしい。こちらは，ミサイルだけでこの大型船を爆破しなければならない。—と，敵は地上のものとはまったく種類を異にした動きをするではないか！この動きで，ラグランジェポイントのベースも破壊されたのだろう。

君はスペースシャトルのミサイル担当管である。ターゲットスコープの中で敵をとらえ，少ないミサイルをむだにせず，地球を守らねばならない。

プログラムリスト

```
10 CGEN2:CGSET1,1:SPRITEON:VIEWSC=0:T=100:X=10:Y=1
0:DIMD(3),X(3),Y(3)
20 FORI=4TO6:DEFMOVE(I)=SPRITE(10,0,2,1,0,1):NEXT
30 VIEW:LOCATE2,0:PRINT"Bﾎﾞﾀﾝ ｦ ｵｼﾃｸﾀﾞｻｲ":IFSTRIG(
0)<>4THEN30
40 LOCATE2,0:PRINT"                  ":DEFMOVE(0)=SPRI
TE(7,3,3,30,0,1):MOVE0:LOCATEX,Y:PRINTCHR$(221)
50 GOSUB100:GOSUB200:GOSUB400
60 IFT=0THEN600
70 GOTO50
100 IFMOVE(0)=0THENL=RND(80)+20:D=RND(8)+1:DX=XPOS
(0)DY=YPOS(0):DEFMOVE(0)=SPRITE(7,D,3,L,0,1):POSIT
ION0,DX,DY:MOVE0
110 RETURN
200 LOCATE X,Y:PRINT" "
205 IFSTICK(0)=1THENX=X+1:IFX=27THENX=0
210 IFSTICK(0)=2THENX=X-1:IFX=-1THENX=26
220 IFSTICK(0)=4THENY=Y+1:IFY=23THENY=2
230 IFSTICK(0)=8THENY=Y-1:IFY=1THENY=22
235 LOCATEX,Y:PRINTCHR$(221)
240 IFSTRIG(0)<>8THEN280
250 I=1:J=0
253 IFMOVE(I)=-1THEN270
260 DEFMOVE(I)=SPRITE(12,1,1,(176-Y*8)/2,0,1):POSI
TIONI,X*8+16,200:MOVEI:X(I)=X:Y(I)=Y:T=T-1:D(I)=1:
J=1
270 IFJ=0THENI=I+1:IFI<4THEN253
280 RETURN
```

★遊び方

このゲームは，ボンバーアタックの少し前の出来事をゲームにしたものです。この高々度衛星軌道上での争いは，敵の圧倒的な物量に対して，地球側はあまりにも間に合わせの武器だったため，大気圏近くでのボンバーアタックの戦いまで，後退を続けるはめになったのです。

プログラムを入力した後，BG グラフィックも入力してください。先にボンバーアタックを入力した人は，その BG グラフィックデーターをそのままこのゲームに使用できます。プレイヤーはコントローラーⅠを使って遊びます。

ゲームを走らせると，Bボタンを押すように指示が出ますので，コントローラーⅠのBボタンを押してください。ゲームが始まります。

画面中で回転しながら動いているのが敵です。十字形をしたキャラクターがターゲットスコープで，このマークを動かすには，ジョイスティックを使用します。画面上部の BEAMの数値はミサイルの残りの数を示し，ミサイルを発射するごとに1つずつ減っていき，残りがφになると，ゲームオーバーです。

ミサイルは一度に3つ発射でき，発射する時にターゲットスコープで設定した位置に達すると爆発します。発射はAボタンを押すことによって行ないます。

点は，遠くで敵を破壊した方がより多くなります。敵はでたらめに方向を変えるので，動きの予想はむずかしいですが，ミサイルの届くまでの時間差のため，ある程度相手の動きを考えなければなりません。

★プログラムの説明

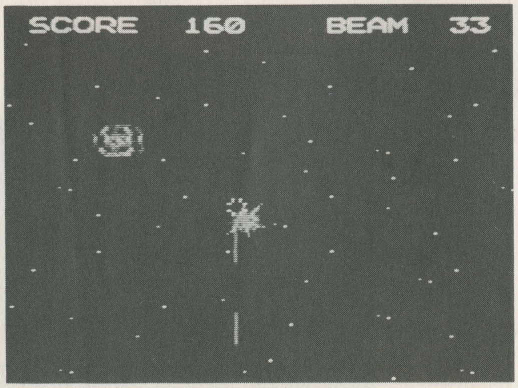
10行からがイニシャライズルーチンです。SC はスコア，Tはミサイルの数です。これを変えれば，ゲームの長さが変わります。

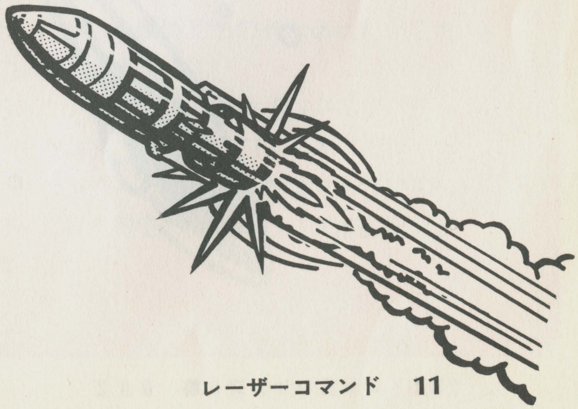
MOVE 4～6は，ミサイルの爆発パターンです。MOVEφが，敵のマザーシップ。この速さを変化させると，最初に出現した時の敵の速さが変わります。

メインプログラムは100行，200行，400行のサブルーチンを呼び出すだけの短いものです。

100行からのサブルーチンは，マザーシップの方向と運動距離を決定するものです。この中のLは運動距離，Dは方向を意味しているので，Lを小さくすると，敵の動きが細かくなります。また，SPRITE 文中の3は，敵の速さのパラメータです。これを2以下にすると，ターゲットスコープは敵に追いつけなくなりますので，ゲームはもっとむずかしくなるでしょう。もし，今のゲームがむずかしいと思うならば，この3を4または5ぐらいにすると良いかも知れません。

```
400 LOCATE1,0:PRINT"SCORE ";SC:LOCATE15,0:PRINT"BE
AM ";T;" "
410 I=1
415 IFMOVE(I)=0ANDD(I)=1THEN430
420 GOTO450
430 ERAI:POSITIONI+3,X(I)*8+16,Y(I)*8+24:MOVEI+3:D
(I)=0
440 IFABS(XPOS(0)-X(I)*8-16)<16ANDABS(YPOS(0)-Y(I)
*8-13)<16THENPLAY"O5C:O4G":ERA0:SC=SC+2*(24-Y(I))
450 I=I+1:IFI=4THEN470
460 GOTO415
470 FORI=4TO6
480 IFMOVE(I)=0THENERAI
490 NEXT:RETURN
600 CLS:SPRITEOFF:LOCATE6,6:PRINT"GAME OVER":LOCAT
E6,9:PRINT"HIT ANY KEY"
610 K$=INKEY$:IFK$=""THEN610
620 RUN
800 LOCATE1,0:PRINT"SCORE ";SC:LOCATE15,0:PRINT"BE
AM ";T; ":GOTO30
```





200行からのサブルーチンは，ターゲットスコープを移動させるものです。この中のDEF MOVEは，ミサイルの発射です。ミサイルの速さ（後の方の）"1"を2や3にすると，ミサイルは遅くなり，ゲームは，とてもむずかしくなります。

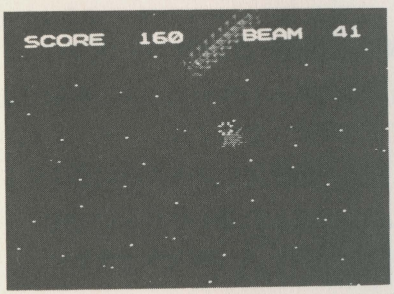
400行からのサブルーチンで，ミサイルの爆発や，敵に当たったかどうかを確認しています。

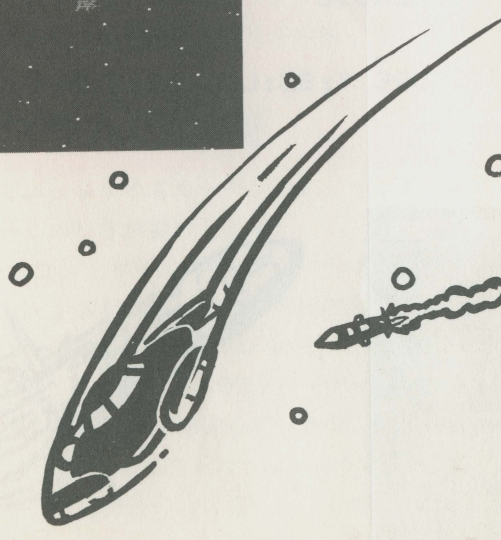
600行からのサブルーチンは，ゲームエンドの時の表示を行なうものです。

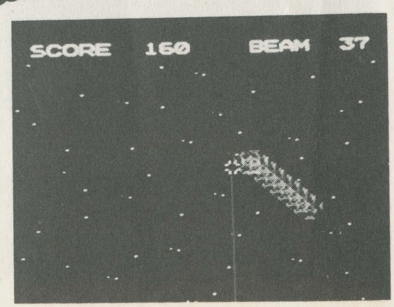
★改造へのヒント

このプログラムでは，敵は攻撃をしかけてきません。メモリー的にまだ余裕がありますから，敵が爆弾を落とすようにしても良いでしょう。ただ，あまり遅くならないようにした方が良いでしょう。その時は，DEF MOVEで，爆弾を設定する時に，味方までの距離を考えたら良いと思います。

この後も，この話のシリーズは続けたいと思いますので，待っていてください。







一口にゲームといっても……

コンピューターゲームにもいろいろある。

まず，インベーダーゲーム以来の「リアルタイム・ゲーム」といわれるもの。ファミコンのソフトにはこれが多いね。弾をうって敵をやっつける「シューティング・ゲーム」と，主人公が歩きまわって何かをする「パックマン型ゲーム」が代表的だ。

「アドベンチャー・ゲーム」といわれるのは，アメリカの名機「アップル」から日本へも上陸してきたゲームで，プレイヤーは主人公になって，ゲーム作者の作った世界を歩いて，いろんな冒険をするものだ。コンピューターをつかった小説ともいえるね。一つ一つの場面を，グラフィックで表示してくれるタイプと，文章だけで情況説明を行うテキストタイプの二つがある。

最近流行しているのが「ロールプレイング・ゲーム」だ。プレイヤーが主人公を演じるのはアドベンチャーと一緒なのだが，このゲームは，ゲームを続ける間に，主人公がどんどん成長する，という点が違う。敵をやっつければ経験値が上がるけど戦闘のため体力が減る。その体力を回復するためには食事をするか休息をとらなければならない。プレイヤーは主人公と共に成長する楽しみがある。

コンピューターがもともと特異とするのは「シミューレーション・ゲーム」といわれる分野だ。

まず，マージャンや将棋・オセロのようなもの。ゴルフや野球のようなスポーツゲーム・カーレースのようなドライブ・シミューレーターや飛行機の操縦のフライトシミューレーター。会社経験のビジネス・シミュレーション。戦争もののウォーゲーム。

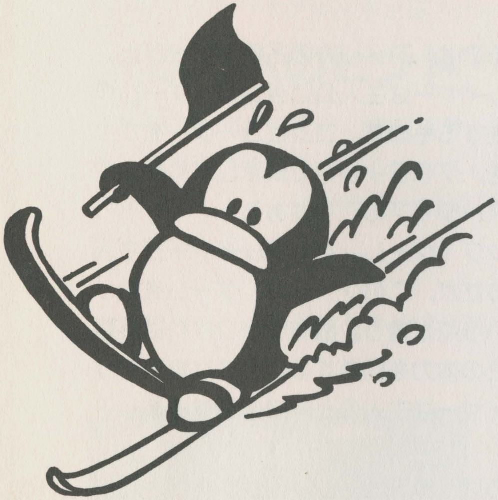
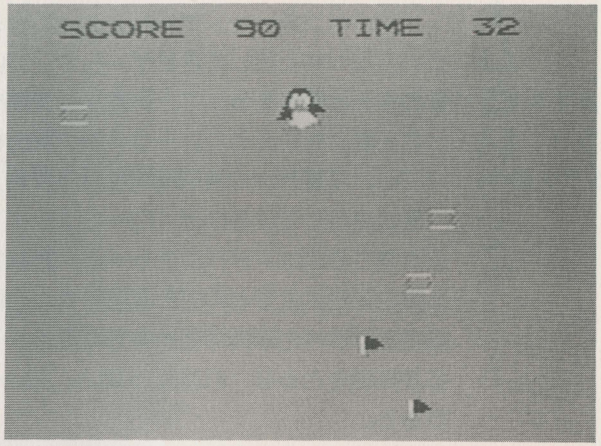
シミューレーションは模技という意味があり，ある意味ではコンピューター・ゲームはみなシミュレーションといえるかもしれない。

「パズル・ゲーム」というのもある。どういう順番でものをとると脱出できるとか，ゲーム自体がパズルになっているものだ。

もっと最近は，「リアルタイム・ロールプレイング」とか「リアルタイム・パズル」とか，いろんな分野の面白い所を取り入れた「クロス・オーヴァー」型のゲームがふえている。あの「ロードランナー」も，敵から逃げるリアルタイムの部分と，どうやったら一面のゴールドを取るか，というパズルの部分がうまくドッキングしたゲームだ。

とにかく，ゲームはおもしろくなくっちゃ！

ペンギンスキー

南極のある長い長いスロープ，観測隊が去っていった後に，人間達が残していった旗があちらこちらに立っています。群からはぐれたペンギンが一羽，この旗を使って遊んでいます。ペンギンは，スラロームで，あちらこちらに残っている旗を通過する。一定時間内に何本通ったかがゲームとなるのです。

★遊び方

プログラムを入力すれば，あとはRUNするだけです。BGグラフィックなんて，いりません。

プログラムを走らせると，画面の下の方から旗や，岩が出て来ます。旗の形をしているのは，ペンギンが通らねばならない旗で，四角いのが，岩。旗を通ると10点もらえますが，岩にぶつかると，時間が短かくなります。

ゲームを行なうには，コントローラーⅠを使ってください。Aボタンを押すと，横方向に移動する量が増え，Bボタンを押すと移動量が減ります。細かい動きをするには，Bのボタンを何回か押した方がいいのですが，旗はそんな事はおかまいなしで出て来るから，１回で大きく移動できた方が良い。まあ，AとBを自由に使いこなすようになるのも，プレイヤーの腕前次第というところでしょうか。

プログラムリスト

```
10 SC=0:T=200:CLS:CGEN2:CGSET0,2:S=3:X=100:SPRITEO
N:PALETB 0,16,22,39,48
20 DEFSPRITE 0,(0,1,0,0,0)=CHR$(104)+CHR$(105)+CHR
$(106)+CHR$(107):DEFSPRITE 1,(0,1,0,1,0)=CHR$(105)
+CHR$(104)+CHR$(107)+CHR$(106)
30 GOSUB100:GOSUB300:GOSUB500:GOSUB200:GOTO30
100 IFSTICK(0)=1THENX=X+S:D=0:IFX<17THENX=X-S
110 IFSTICK(0)=2THENX=X-S:D=1:IFX>220THENX=X+S
120 SPRITED,X,50:IFD=0THENSPRITE1:GOTO130
125 SPRITE0
130 IFSTRIG(0)=8THENS=S+1:IFS=10THENS=9
140 IFSTRIG(0)=4THENS=S-1:IFS=0THENS=1
150 RETURN
200 IFRND(64)<12THENLOCATERND(25)+1,23:PRINTCHR$(2
09)
210 IFRND(64)<7THENLOCATERND(25)+1,23:PRINTCHR$(20
9)
220 IFRND(64)<10THENLOCATERND(20)+3,23:PRINTCHR$(1
99)
230 LOCATE27,23:PRINT" ":RETURN
300 D1=(X-16)/8:D2=D1+1:A1=ASC(SCR$(D1,4)):A2=ASC(
SCR$(D2,4)):A3=ASC(SCR$(D1,3)):A4=ASC(SCR$(D2,3))
310 IFA1=199ORA2=199ORA3=199ORA4=199THENSC=SC+10:P
LAY"O5E"
320 IFA1=209ORA2=209ORA3=209ORA4=209THEN400
330 T=T-1:IFT<1THENRETURN600
340 RETURN
400 BEEP:T=T-5:IFT<1THENRETURN600
```

実際にペンギンを動かす時は，コントローラーIのジョイスティックの右と左を使ってください。行きたい方向のキーを押すと，ペンギンはその方向へとすべって行く。この動きの大きさを，AとBで決めることは，もうわかったでしょうね。

時間は200秒，この間に何個の旗をクリア出来るか！　200点も出るようになれば，君のペンギンは，もはや，プロスキーヤーです!!

★プログラムの説明

そもそもアクションゲームには，色々な種類がありますが，キャラクターより下の画面全体が一定方向（何も一方向とは限りません。）に動くようなゲームをスクロールゲームと呼びます。

BASICで画面をスクロールさせるには，1個ずつ画面をずらしていたのでは，時間がかかりすぎてゲームになりません。そこで，よく使われる方法で，BASICテキストのスクロールを使用するやり方があります。つまり，画面の右下に何かを描くと，自動的に画面全体が一段上にスクロールしてしまう性質を利用して，ここに何か（このプログラムでは“　”）を表示して，画面を上の方に流していくのです。

ペンギンスキーでは，スプライト画面に描かれたペンギンを，ジョイスティックで動かし，画面の一番下の行に岩，旗を描くことで画面の構造物を作っています。実際のプログラムは以下のようになっています。

●初期設定（10～　）

T：時間。これが1回動くごとに1減ってゆき，φになるとゲームオーバーとなる。

SC：点数100行から—コントローラーの読み込み。

S：1回に動く量。このSを大きくすると，早く動くようになる。

●岩，旗の発生（200～　）

CHR$（199）は旗，CHR$（209）は岩，これをRNDで発生させているので，このIF文の数値を変えると，難易度が大きく変化する。

●地目のチェック（300～　）

このルーチンで，ペンギンが，岩や旗にぶつかったかどうか調べている。

●岩につまづいた場合（400～　）

T（時間）を5だけ減らしている。ここでTをより大きく減らすようにしても，おもしろくなる。

●画面の上端にスコアと時間を表示する（500～　）

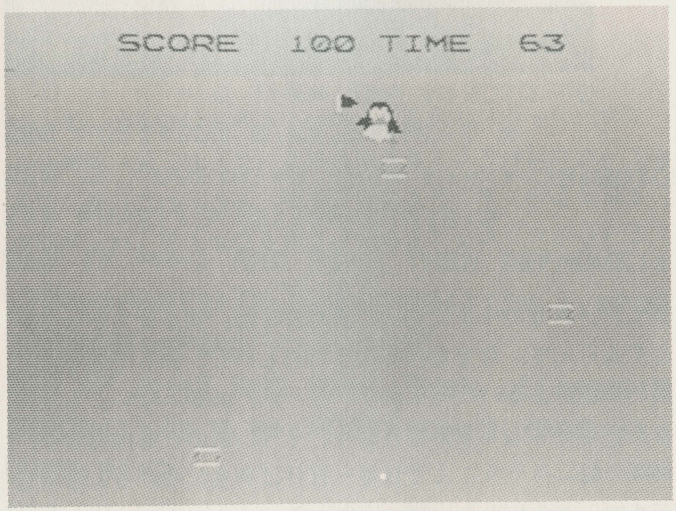
●タイムオーバー（600～　）

このプログラムで，スクロールゲームのやり方がわかったと思います。背景を空から見た

```
410 RETURN
500 LOCATE3,0:PRINT"SCORE ";SC:LOCATE14,0:PRINT"TI
ME ";T:RETURN
600 PLAY"O5CG:O3EG":CLS:LOCATE6,6:PRINT"TIME OVER"
:LOCATE6,8:PRINT"HIT ANY KEY":LOCATE6,10:PRINT"SCO
RE ";SC
610 K$=INKEY$:IFK$=""THEN610
620 RUN
```

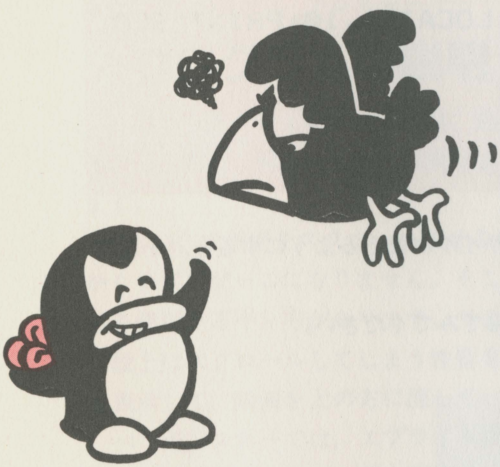
地面にして，ペンギンを飛行機にします。そして，ミサイルが使えるようにすると，それでもう立派なスクロール・アクションゲームとなります。

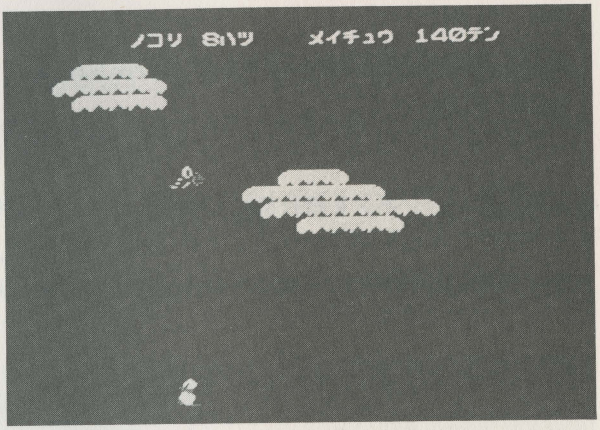
みなさんで，それぞれプログラムの改造をして楽しんでみてください。





ペンギンさんとからすさん





最近はずいぶん有名になって，コマーシャルには出るわ，ぬいぐるみにはなるわ，まんがなどにもずいぶん出演するわ，ふとったつばめさんとまちがわれる（？）わ，遠い南極からわざわざ出てきたかいがあった，と言えるペンギンさんなのですが，ふと，われに返ってみるとなんとなくむなしいのです。どうしたらこんなにもむなしくなるのかと思うほど気がめいってしまうのです。

そこで，ペンギンさんは考えました。自分の今までやってきたことは，何かまちがっていたのだろうか？　人々にたのしんでもらって，自分もけっこうたのしいし，決して悪いことではないはずだ。それでは何か別の問題があるのだろうか？　うーん，わからない。もう一度，あのふるさと南極から出てきたときからよく考えてみよう。そう，あのときはたしか，有名になりたい一心で，南極観測隊の船にもぐりこんで日本へ行けば有名になれるんじゃないかと思っていたんだ。そしてどうにかこうにか日本についたときはなんともいえずうれしかった 。それからいろんなことをやっているうちに，いつのまにかテレビに出ることになり，あっという間に有名になったんだった。しかし最近なぜかむなしい。いったいこれはなんんだ。などと考えながらふと窓の外を見ると，電線にはすずめさんたちが，電柱にはからすさんが，とまっていました。すずめさんたちは，なにかぺちゃくちゃ話しているし，からすさんは空を見上げてじっとしていました。しかし突然すずめさんたちの話しがまとまったの

プログラムリスト

```
5 CLS:VIEW:SPRITE ON:CGSET1,0
10 NO=30:GOSUB 1000
80 DEFMOVE(1)=SPRITE (4,3,1,60,0,0):DEFMOVE(7)=SPR
ITE (10,1,9,3,0,0):DEFMOVE(3)=SPRITE(12,1,1,100,0,
0):POSITION 1,120,210:PI=1:MOVE1
90 HO=1:KX=RND(100)+70:KY=150:DEFMOVE(0)=SPRITE(15
,7,1,KX/2,0,0):GOTO140
100 HO=HO+1:IF HO=2 THEN HO=0
120 DEFMOVE(0)=SPRITE(15,(HO+1)*4-1,1,120,0,0)
140 IF KY=150 THEN 190
160 IF HO=1 THEN KX=240:GOTO190
180 KX=0
190 IF KY=10 THEN ERA 0:GOTO 90
200 POSITION 0,KX,KY:MOVE 0
220 IF MOVE(0)=0 THEN KY=KY-20:GOTO100
240 PH=STICK(0):PB=STRIG(0)
260 IF PH=1 AND PI=2 THEN DEFMOVE(1)=SPRITE (4,3,1
,120-XPOS(2)/2,0,0):POSITION 1,XPOS(2),YPOS(2):ERA
 2:MOVE1:PI=1
280 IF PH=2 AND PI=1 THEN DEFMOVE(2)=SPRITE (4,7,1
,XPOS(1)/2,0,0):POSITION 2,XPOS(1),YPOS(1):ERA 1:M
OVE2:PI=2
300 IF MOVE(3)=0 AND PB=4 THEN DEFMOVE(3)=SPRITE (
12,1,1,100,0,0):POSITION 3,XPOS(PI),YPOS(PI):MOVE
3:NO=NO-1:GOSUB 1000
320 AX=XPOS(3)-XPOS(0):AY=YPOS(3)-YPOS(0)
330 IF MOVE(3)=0 THEN ERA 3:IF NO<1 THEN 600
```

でしょう。飛び立っていってしまいました。

別になんでもない，いつものことなんですが，ペンギンさんは，大へんなショックを受けてしまいました。そうです，ペンギンさんは，鳥であるにもかかわらず，こともあろうに空を飛ぶことができないのです。それはもちろん南極の仲間たちはペンギンだからみんな空は飛べません。そんな仲間たちといたときは，当然空を飛べないなんて気にもなりません。

しかし，この日本じゃ飛べない鳥なんていません。にわとりさんにしたって少しですけど飛べますし，あとはせいぜい，だちょうさんか，ヤンバルクイナさんぐらいでしょう。

それにひきかえ，ペンギンさんはまったく飛べないうえに，足ひれまでついています。ペンギンさんは，飛べさえすればこのむなしさも消えるだろうと思い，まだ電柱の上のからすさんに，飛び方を教わろうといそいで外へ飛び出しました。しかし，足のおそいペンギンさんです。ペンギンさんが電柱の下へつくころには思いにふけっていたからすさんも飛び立ったあとでした。

ペンギンさんは，遠くに飛んでいくからすさんを見ながら，プライドの高いからすさんは飛べない鳥は相手にしないのだと思いこみ，からすさんがねたましくなってきました。

からすさんだってあの羽根さえなければ飛べやしないんだ。よし，石をぶつけて羽根を折ってしまえばいいんだ。

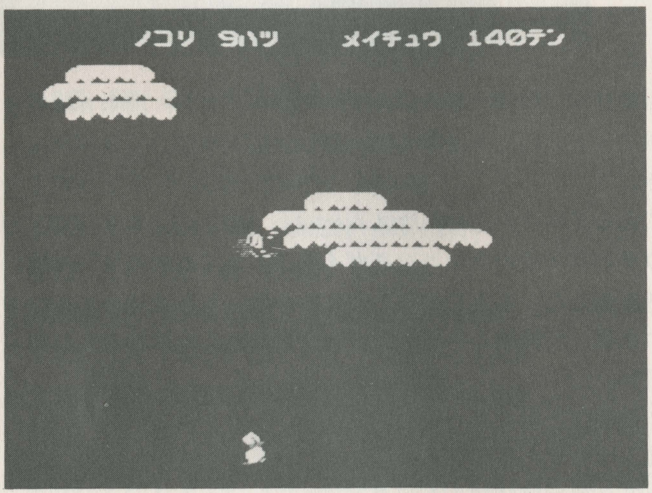
と，いうわけで，ペンギンさんのからすさん攻撃が始まりました。

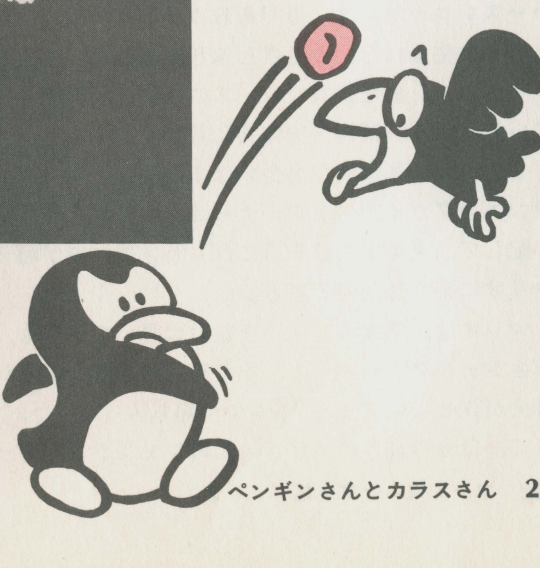
★ゲームの説明

ゲームは，RUNするとすぐに始まります。近い所からジグザグに，空高くからすさんが飛んで行きます。それをコントローラーＩのジョイスティックで左右に動かし，トリガーボタンBで石を投げます。点数は上の方で当るほど点数が高くなります。石は30個しかありません。すべて投げつくすとゲームは終わりです。

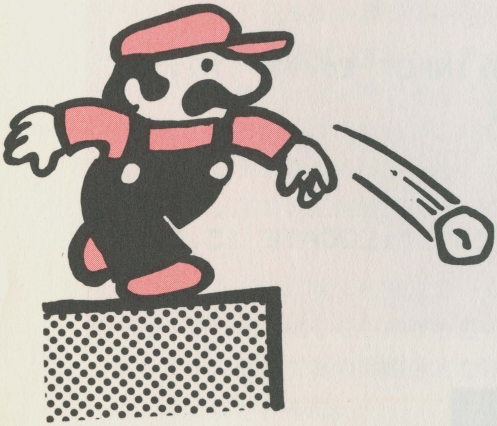
また，BGグラフィックで雲などを書いて，障害物とするのもよいでしょう。その場合は，5行のCLSとSPRITEの間にVIEWを入れてください。また，プログラム中のDEF MOVE (Ø)＝⊡⊡⊡の所の表示優先度を1にすると，からすさんが雲にかくれたりします。一つ例をあげておきますが，みなさんで好きなように書きかえてください。さらにMOVEのスピードを変えるのもおもしろいでしょう。

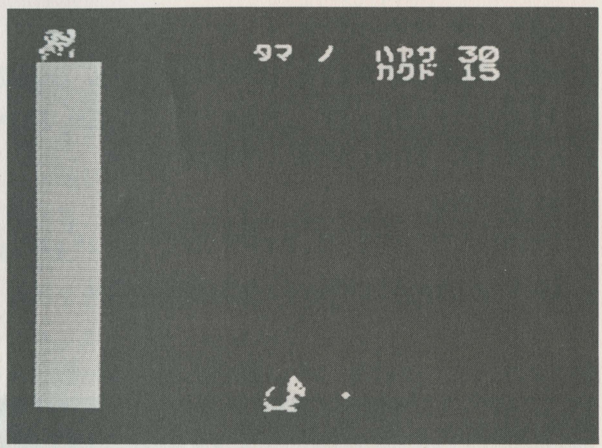
```
340 IF AX>-4 AND AX<12 AND AY>-4 AND AY<12 THEN CU
T 0,3:ERA 0,3:BEEP:POSITION 3,0,0:ME=ME+(170-KY)/2
:GOSUB 1000:GOTO 90
360 GOTO 220
600 ERA 0,1,2
601 IF STRIG(0)<>4 THEN 600
605 CLS:SPRITE OFF:LOCATE 5,10:INPUT"ﾓｳｲﾁﾄﾞ ﾔﾘﾏｽｶ
(Y/N)";TR$:IF TR$="Y" THEN RUN
610 IF TR$<>"N" THEN 605
620 END
1000 LOCATE 5,0:PRINT"ﾉｺﾘ";NO;"ﾊﾂ ":LOCATE 15,0:PR
INT"ﾒｲﾁｭｳ";ME;"ﾃﾝ":RETURN
```





逃げるなカメよ！





名前，マリオ・デ・マリオ。年齢32才。職業，競ガメのトレーナー。過去，国際A級競カメ選手権大会で入賞４回，同Ｂ級で入賞３回，準優勝１回，但し最近６年は，トレーナーとしての活動は休んでいる。酒好き。

名前，シェルクリーパー。父であるアマゾン産のハウスムーバー（第３回全米オークスでコースレコード）と，母であるガラパゴス産のランニングガイア（あの“南米の星”の娘）との間に生まれた。大きさと速さを備えた，まるで走るためにこの世に生を受けたようなエリートカメ。まだ正式なレースに出場した経験こそ無いが，末公認ながら世界新を３つ持っていると云う。精神，肉体共に若い。

この二人の運命的な出合いは２週間ほど前にあった。たまたま，シェルクリーパーの母，ランニングガイアのトレーナーは，マリオの恩師であったため，マリオの酒びたりの生活を心配して，ガイアの息子（これも若さ故に壁が超えられずにいた）シェルクリーパーを，マリオに紹介したのである。

マリオは，恩師の心使いを重荷に思いながらも，トレーナーとしての再出発を夢見ていた。若きシェルクリーパーは，その天性の才能と家系が災いして，努力することを嫌い，そんな自分が許せないまま，ジレンマに陥ち入っていた。マリオは，そんなシェルクリーパーの姿を見て，自分の弱さに気付き，このシェルクリーパーをジレンマから抜け出させ，レースで勝

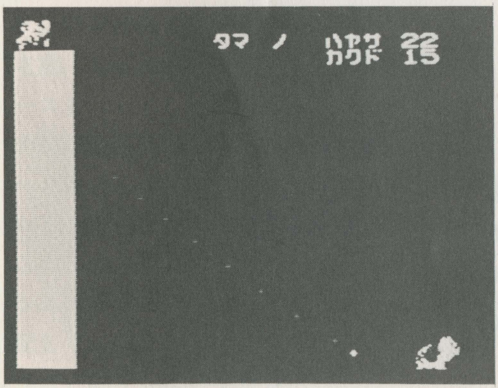
プログラムリスト

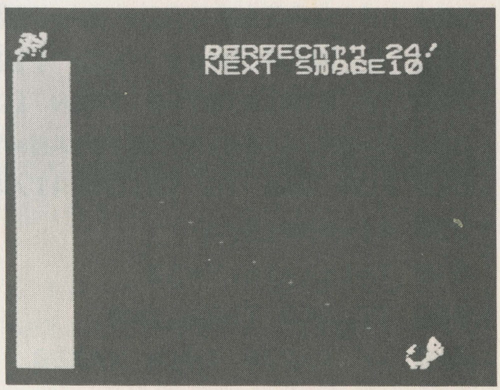
```
90 SE=41
100 CLS:VIEW:SPRITE ON:DEF SPRITE 1,(0,0,0,0,0)=CH
R$(208):DEF MOVE(0)=SPRITE(13,3,SE,204,0):MOVE0
102 DEF SPRITE 0,(0,1,0,1,0)=CHR$(13)+CHR$(12)+CHR
$(15)+CHR$(14):SPRITE 0,16,24
105 POSITION 0,32,176:MOVE 0
110 SP=0:AN=0:X=24:Y=32:G=0
120 LOCATE10,1:PRINT"ﾀﾏ ﾉ  ﾊﾔｻ";SP;"    ";:LOCATE10
,2:PRINT "       ｶｸﾄﾞ";AN;"  ";:G=0
130 ST=STICK(0):IF ST=0 THEN210
140 IF ST=2THENVX=-2:VY=0:GOTO160
141 IF ST=1THENVX=2:VY=0:GOTO160
142 IF ST=8THENVY=1:VX=0:GOTO160
143 IF ST=4THENVY=-1:VX=0:GOTO160
160 SP=SP+VX:AN=AN+VY
170 IF AN<0 THEN AN=0
180 IF AN>90 THEN AN=90
190 IF SP<0 THEN SP=0
200 IF SP>100 THEN SP=100
210 IF MOVE(0)=0 THEN 1100
220 IF STRIG(0)=0 THEN 120
300 V=SP/4+50:RT=810+AN*AN:A=2
310 FOR B=0 TO 10:DT=RT/A:A=(DT+A)/2:NEXT
320 SI=AN*10/A
330 T=1:X=16:Y=32
400 IF X<0 THEN X=0
410 IF X>230 THEN X=230
420 IF Y<0 THEN Y=0
```

たせてやることで，自分自身もまた，新しい生活を始められると思うのである。
その日もまた，きびしいトレーニングに飽きていたシェルクリーパーは，ついに「もう，こんな生活はいやだ！」と言うと，マリオのもとを去ろうとした。温室育ちのシェルクリーパーは，一度なぐらなければいけない。もうマリオの追いつける距離では無い。しょうがなくて，マリオは石を投げる。「シェルクリーパーよ，怒っても良い。気付いてくれ！」
だが，石が当たらねば何にもならない。
酒のせいでコントロールのきかないマリオに替って，君が石を投げてくれないか。
青春は，自ら選んだ挫折の道であってはならないのだから…。

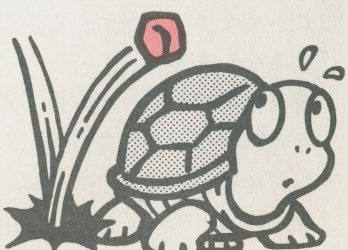
★プログラムの説明

ゲームが始まると，右上に球の速さと角度の表示が出ます。スティック（I）を使って両方の数を変化させ，トリガーボタン（どれでもよい）で球を放ってください。カメに球をあてれば成功です。450行のXとYの式をいろいろ変えてみると，球の動き方が変わります。ぜひプログラムを作りなおしてみてください。
角度はAN，スピードはSPという変数に入っています。この数をXとYの値にいろいろな方法で加えて，球の動きに変化を付けるとよいと思います。
カメのスピードは，だんだん早くなってきます。最高速のカメをやっつけられたら，「PERFECT」の表示がでます。君はそこまで行きつけるかな？
90行と2200行のＳＥがカメのスピードです。いろいろ変えてみてください。

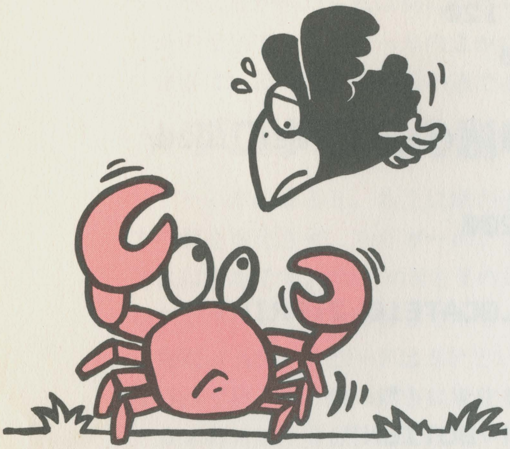


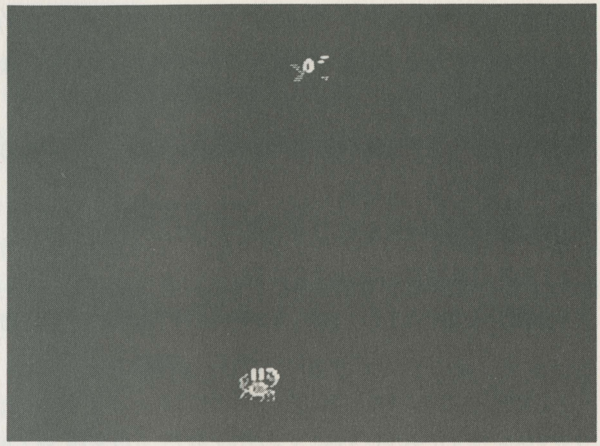


```
430 IF Y>196 THEN Y=200
440 SPRITE 1,X,Y
450 X=X+1+V*T/200:G=(AN*2+SP+T)*50/600:Y=Y+G
455 IF MOVE(0)=0 THEN 1100
460 IF Y<0 THEN Y=0
465 IF X>229 THEN SPRITE 1:GOTO 120
470 IF Y<185 THEN T=T+1:GOTO 400
480 SPRITE 1
490 IF XPOS(0)>(X-16) AND XPOS(0)<(X+21) THEN 1000
510 GOTO 120
1000 A$="HIT !!          ":GOTO 2200
1100 BEEP:A$="FAIL!!         "
1200 LOCATE 10,1:PRINTA$:ERA 0:LOCATE10,2:PRINT"
            ";:PAUSE 300
1295 LOCATE10,2:PRINT"                 "
1300 LOCATE10,1:PRINT "HIT START BOTTUN"
1400 IF STRIG(0)<>1 THEN 1400
1500 RUN
2200 SE=SE-4:IF SE<1 THEN A$="PERFECT     !"
2300 LOCATE 10,1:PRINTA$:ERA 0:IF SE>0 THEN LOCATE
10,2:PRINT"NEXT STAGE ! ";:PAUSE 300:GOTO 100
2350 LOCATE10,2:PRINT"                 "
2400 PAUSE 400:GOTO 1300
```



クロウ来るな!





その昔，神様がこの世界をお作りになってからしばらくは，世の中はとても静かで平和だったそうな。生物はみな１つの所にいても争わず，仲たがいもせず暮らしていたと。

だが，そんな中にも意地悪な者もいて，回りを悩ませていたんだと。

カラスもそんな一人で，いたずらが過ぎるのでまわりから煙たがられていた。カニさんも息子のハサミを折られた上，家の回りを荒されたりもしたそうな。

物に動じないカニさんも，とうとう怒り出した。地面を走り回ると，ハサミをふり立てて大声を出したんだと。

けれども，そんなことでまいるカラスではなかったな。すきを見つけては，舞い降りて来る。さすがにカラスの真下に来てハサミをふり回すと，カラスは空高く逃げる逃げる。

ほれ，またカラスが飛んで来る。

「来るな，来るな！」

「降りたらだめだ。カンカンカラスのくちばしよりも，こっちのハサミが堅いぞ。」

その後，こんなささいな出来事がもとで，生き物はけんかをした。そして空を飛べる者は空へ，泳げる者は海や河へと，その住みかを分かち，今日のようになったのだそうな。

今は，知る者とて少ない，遠い遠い昔のことじゃ。

プログラムリスト

```
100 CLS:SPRITEON
110 DEF SPRITE0,(0,1,0,1,0)=CHR$(200)+CHR$(201)+CH
R$(202)+CHR$(203)
120 DEF SPRITE1,(0,1,0,0,0)=CHR$(204)+CHR$(205)+CH
R$(206)+CHR$(207)
130 DEF SPRITE2,(2,1,0,1,0)=CHR$(192)+CHR$(193)+CH
R$(194)+CHR$(195)
140 DEF SPRITE3,(2,1,0,0,0)=CHR$(196)+CHR$(197)+CH
R$(198)+CHR$(199)
150 LOCATE10,1:PRINT"HIT ANY KEY";:PAUSE
160 LOCATE10,1:PRINT"            ";:PY=2
170 SPRITE ON:X=16:Y=20:PX=32:PY=220:P1=2:P2=3:M1=
0:M2=1
200 PAUSE5:GOSUB2000:SPRITE M1,X,Y:SPRITE P1,PX,PY
:SPRITE P2:SPRITE M2
210 IF Y>PY-2 THEN 1100
220 IF X>220 THEN 1000
230 X=X+1:Y=Y+DY:DY=DY+1:ST=STICK(0)
234 IF ST=0 OR ST>3 THEN 238
235 ST=-2*SGN(-3+ST*2):GOTO 240
238 ST=0
240 PX=ST+PX:GOSUB 2000
250 IF STRIG(0)=0 THEN 200
260 IF (PX-10+X/70)>X OR (PX+10-X/70)<X THEN 200
270 DY=DY-2-X/100:SWAP M1,M2:SWAP P1,P2:GOTO 200
1000 A$="ｾｲｺｳ":PO=1000-4*Y:GOTO 1200
1100 A$="ｼｯﾊﾟｲ":PO=0
1200 LOCATE10,1:PRINT A$;"ﾄｸﾃﾝ";PO;
```

★遊び方

プログラムを打ち込んで，RUNすると，ゲームが始まります。HIT ANY KEYと表示が出るので，キーボードのキーを何か押してください。画面が変わって，カラスとカニさんが登場します。プレイヤーが扱うのは，コントローラーＩのジョイスティックの右方向，左方向，およびトリガー全部です。ジョイスティックの右，左を押すと，カニさんが右，左に動きます。トリガーは，どれを押しても動作は同じで，このトリガーを押すと，カニさんが怒ります。

カラスさんは，画面の左上に出現した後，右へ右へと飛びながら，ともすればカニさんのいる下の方へ来ようとします。カニさんをカラスさんの真下へ動かして怒らせると，しかたなくカラスさんは，パタパタとはばたいて，上空へ昇っていきます。

こうしてカニさんを扱いながら，カラスさんを地面に降ろさず，画面の右端まで追いやってください。画面の右端にカラスさんが着いた時点で，点数が出ます。カラスさんが高ければ高いほど，点が高く，カラスさんが低いと，点も低くなります。点は，最高で1000点です。途中でカラスさんを地面に降ろしてしまうと，GAME OVERとなります。うまくカニさんを動かして，カラスさんを追っ払ってください。

★プログラムの説明

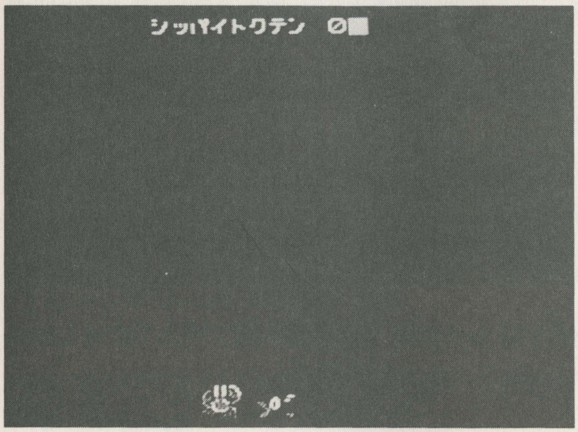
このプログラムのキャラクターは，すべてスプライトで表示しています。X，Yがカラスの位置，PX，PYがカメの位置です。カラスはカメと縦の位置が約1キャラクター分のはばの中で重なっている時にトリガーが押されていれば，はばたいて上に行こうとします。この部分は，PXとXの差が約2キャラクター分以内にあること。つまり差がある数以内であるかどうかをIF文で判断しています。この差を変えることによってこのゲームをもっとむずかしくすることも出来ます。下の○の部分を変えてみてください。

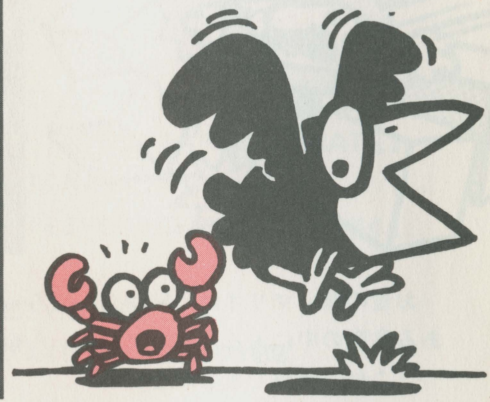
260　IF（PX＋○）＞X　OR（PX＋○）＜XTHEN～

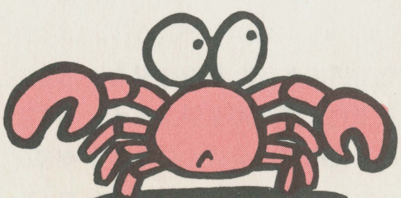
カラスの動き方をよく見てください。下に落ち始めると急に速くなっていますね。これはDY　という変数の値にポイントがあります。DYは下に動くと加算されます。DYは1回動くごとにYに足されるので，落下にしたがって落ちる速さも速くなるのです。これは重力がある場面，つまり宇宙でなく地上などでの物の動きをリアルにする時によく使います。

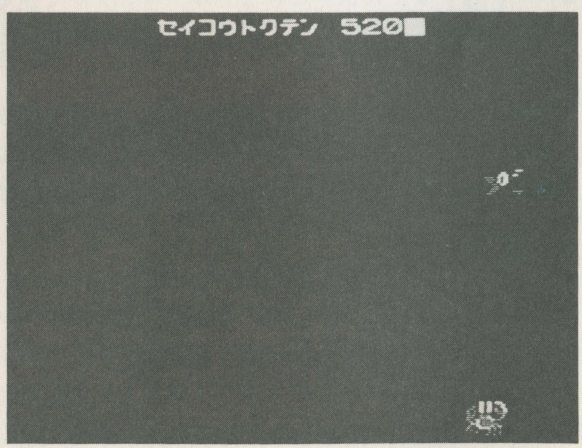
ファミリーコンピューターは小数を使えないので，計算のプログラムはやっかいです。計算結果は切りすてられることにいつも注意しましょう。掛け算と割り算がいっしょになっている時には，先に掛け算をやって，割り算は最後にやるようにすると，結果が良いでしょう。

```
1300 PAUSE
1400 RUN
2000 IF PX<0 THEN PX=0
2100 IF PX>230 THEN X=230
2200 IF Y<0 THEN Y=0
2300 IF Y>230 THEN Y=230
2400 RETURN
```

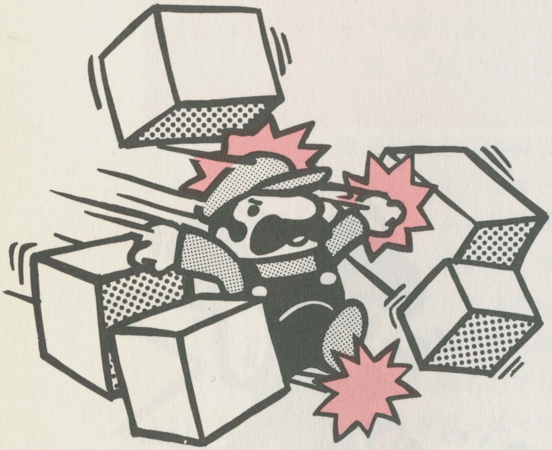


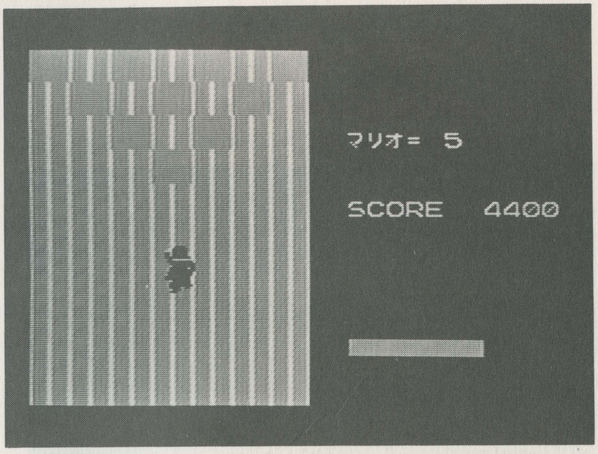






ボーリング





お金の無いマリオは，何かおもしろい遊びがないかと，あたりをブラブラしていました。ある倉庫の中に入ると空のダンボール箱が10個見つかりました。

「これで何ができるかな」

積木をするのはつまらないし，パズルをやるのは頭がいたい。絵を描こうにもペンがないし……。待てよ。10個。これならボーリングができる。ボールは僕がなればいいや。

マリオは箱を床の上にならべると，自分が球のかわりに走っていって，ダンボールをけちらします。

10個のダンボールが全部どけられると，もう1度並らべ直し，また突進していきます。マリオは10回走り，何個の箱をどけられたかをゲームにしました。

床がツルツルですべるので，変に力を込めて斜めに走ると，すべって転んでしまいます。

★遊び方

プログラムを打ち込み，BGグラフィックも打ち込んで，プログラムを走らせると，ボーリングのピンに相当するダンボールが表示され，画面の左下にマリオが登場します。

ここでマリオはコントローラーIのジョイスティックの左または右を押すことによって，動きます。このマリオは，走り出す点を決めるものです。

プログラムリスト

```
10 SPRITEON:VIEW:CGEN2:CGSET1,0:DIMX(9),Y(9),Z(9),
XX(1):B=10:SC=0:DEFSPRITE2,(0,1,0,1,0)=CHR$(21)+CH
R$(20)+CHR$(23)+CHR$(22):DEFSPRITE3,(0,1,0,0,0)=CH
R$(20)+CHR$(21)+CHR$(22)+CHR$(23)
20 DEFMOVE(0)=SPRITE(0,3,1,1,0,0):DEFMOVE(1)=SPRIT
E(0,7,1,1,0,0):DEFMOVE(2)=SPRITE(12,3,1,48,0,0)
30 RESTORE999:FORI=0TO9:READX(I),Y(I):NEXT
35 FORI=0TO9:Z(I)=1:NEXT
40 LOCATE16,5:PRINT"HIT A":FORI=0TO9:GOSUB900:NEXT
50 X=20:Y=200:Z=0:LOCATE16,5:PRINT"ﾏﾘｵ=";B:LOCATE1
6,9:PRINT"SCORE ";100*SC:POSITION0,X,Y:MOVE0
60 FORI=0TO1:POSITIONI,X,Y:NEXT:IFSTICK(0)=1ANDX<1
12THENMOVE0:ERA1:Z=0
70 IFSTICK(0)=2ANDX>20THENMOVE1:ERA0:Z=1
80 X=XPOS(Z):IFSTRIG(0)=8THENXX(0)=X:GOTO100
90 GOTO60
100 POSITION2,16,160:MOVE2
110 IFSTRIG(0)=4THENXX(1)=XPOS(2):ERA(2):GOTO200
120 IFMOVE(2)=0THEN100
130 GOTO110
200 Z=16:S=0
210 LOCATEZ,17:PRINTCHR$(254)+" ":IFZ<17THENZZ=1:G
OTO230
220 IFZ>25THENZZ=-1
230 IFSTRIG(0)=8THENS=Z-16:GOTO300
240 Z=Z+ZZ:GOTO210
300 P=(XX(1)-XX(0))/8:ERA0,1
310 SPRITE2,X,Y:GOSUB400:SPRITE3,X,Y:GOSUB400:IFY<
```

走り出す点が決まったらAボタンを押し，その点を設定します。Aボタンを押した後は，マーカーがマリオの少し上の点を，左から右へ移動していきます。Bボタンを押すと，マリオが走って行く方向が設定できます。マリオは，Aボタンを押した点から，Bボタンを押した時にマーカーの示していた方向へと走るようになっています。この後，画面の右下に，マリオの走る力を示す棒が現われ，伸びたり縮んだりします。この力は，Aボタンを押して決めます。

Aボタンが押された時に画面に表示されている力で，マリオはあらかじめ設定したように走っていきます。マリオが直接触れたダンボールと，そのダンボールによってぶつかられたダンボールは，画面から消されます。

10個のダンボールが全部消えると，箱はみな並べ直されます。マリオが１回走るごとに，画面右上のマリオの数が減っていき，マリオが０になると（つまり，マリオが10回走ると），ゲームオーバーになります。続いてゲームをやりたい時は，画面のメッセージにしたがって何かキーを押して下さい。

マリオの立つ場所，走っていく方向，力によってマリオはすべり，方向が変ってしまいます。この３つをいろいろと試してみて，ストライクを出してください。

★プログラムの説明

●**10行**からがイニシャライズルーチンです。SPRITEの２および３が走って行くマリオ，MOVEの０，１が右向き，左向きのマリオのキャラクターです。ここを変えればレイディのボウリングにも，ペンペンのボーリングにもなります。

●**50行**からが，マリオの位置決めルーチン，**100行**からのルーチンが，方向決めです。**200行**からのルーチンが，力を決定するものです。この中のSが，他のルーチンに引きつがれる強さを示す変数です。

●**300行**からのロジック中で，マリオの走り方が決定されます。Pは曲がり方を決定する変数です。これを大きくすると曲り方が急になります。また，SおよびXを決定している式を変化させると，曲がる位置なども変化します。これらを変化させて，ストライクが出やすくしたり，直球しか出ないようにすることも可能です。いろいろと試してみてください。

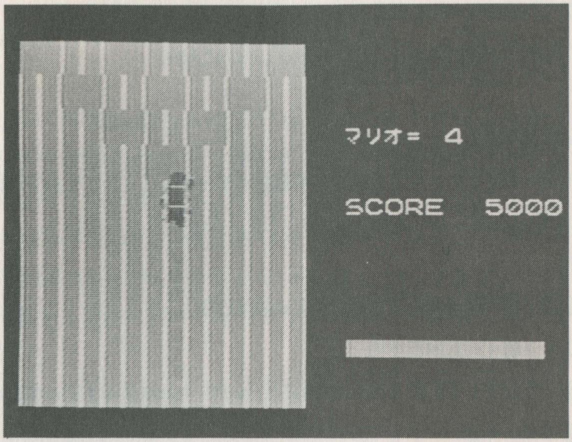
●**600行**からは，マリオが直接触れなかったダンボール箱が，倒れるかどうかを決定するルーチンです。この中のIF文を変えると，マリオがまっすぐ走っていった場合でも，全部倒れるようにもなります。

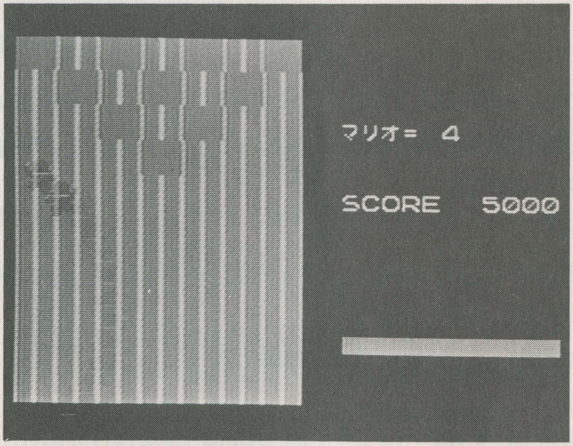
●**900行**からのルーチンは，ダンボール箱が置かれる位置を計算するものです。その計算のもとになるX（I），Y（I）の中に入っている数値は，999行のDATA文の中身です。

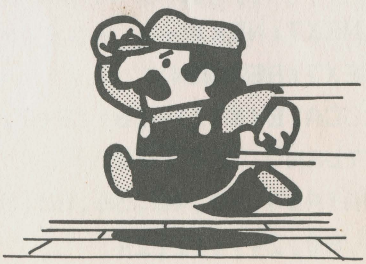
今回は，マリオの動き方を複雑にしたために，メモリーの余裕がなくなってしまったので，

```
24THENGOTO500
320 GOTO310
400 FX=P+X-(200-Y)*P/50 :Y=Y-S
410 IFFX<0THENX=0:RETURN500
420 IFFX>128THENX=128:RETURN500
430 X=FX
440 FORI=0TO9
450 IFABS(X-X(I))<14ANDABS(Y-Y(I))<16THENZ(I)=0
460 NEXT
470 RETURN
500 SPRITE2:SPRITE3:FORI=0TO9
510 IFZ(I)<>0THEN540
520 FORK=I TO9:IFZ(K)=1ANDX(K)>=X(I)-16ANDX(K)<=X(
I)+16THENZ(K)=2
530 NEXT
540 NEXT
550 C=0:FORI=0TO9:IFZ(I)=0ORZ(I)=2THENC=C+1
560 NEXT:SC=SC+C:VIEW:GOSUB900:B=B-1:IFB=0THEN600
570 IFC=10THEN35
580 GOTO50
600 SPRITEOFF:LOCATE16,7:PRINT"GAME OVER":LOCATE16
,11:PRINT"HIT ANY KEY"
610 K$=INKEY$:IFK$=""THEN610
620 RUN
900 FORI=0TO9:IFZ(I)=1THENFORJ=0TO1:FORL=0TO1:GOSU
B950:NEXT:NEXT
910 NEXT:RETURN
950 LOCATE(X(I)-16)/8+J,(Y(I)-24)/8+L:PRINTCHR$(25
4):RETURN
999 DATA64,72,48,56,80,56,32,40,64,40,96,40,16,24,
48,24,80,24,112,24
```

スコアの正確なロジックは入れてありません。みなさんが，プログラムを縮めて，スコア計算のプログラムを乗せてください。







コンピューターむかしむかし

もともとコンピューターは科学技術計算とか，残念なことに，軍隊が使ったりするために開発されたといってもいい。大きなビルの一フロア全部を占めたコンピューターにはテレビなんかつながってなく，大型で音が大きいタイプライターとプリンターしかなかった。だいいち，コンピューターを一分間使うだけで何百ドルもかかっていた。コンピューターでゲームをするなんて，とても考えられる状態じゃなかった。

「TRON」という映画を見ただろうか。人間がコンピューターの中に入っていくやつだ。みんなはあの映画を見てどう思ったかな。実は，昔はそれがあたりまえだったんだ。

つまり，今のコンピューターはＩＣやＬＳＩを使ってるね。あのＩＣやＬＳＩっていうのは真空管というものを何万・何十万個をつめこんだのと同じ働きをしているんだ。当然，昔はＩＣやＬＳＩなんてないから，コンピューターの内部には何十万個も真空管がならんでいた。（もう少したつと，真空管のかわりにトランジスタが使われだすんだ。）

ところでこの真空管，細長い電球みたいな形をしてるんだが，電球と同じように，よく切れるんだ。切れた真空管をとりかえるには，コンピューターの内部へ人間が入っていって手でとりかえるわけなんだ。

そのころのScience Fictionには，科学技術の進歩した21世紀には，コンピューター内部を歩きまわってこわれた真空管をとりかえるロボットをもった自動修復機能つきの超コンピューターなんかも登場してくる。

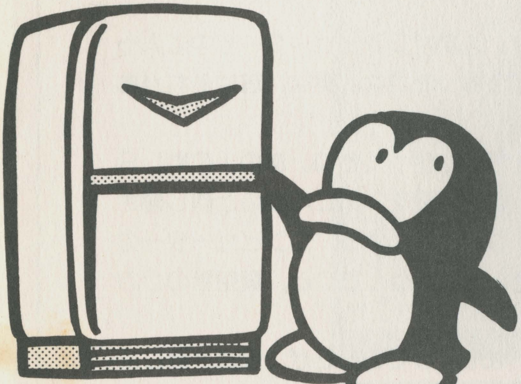
今だと，故障したと思われるＬＳＩを組み込んだ板（基板）ごと，ポンと入れかえるだけ，簡単なもんだ。

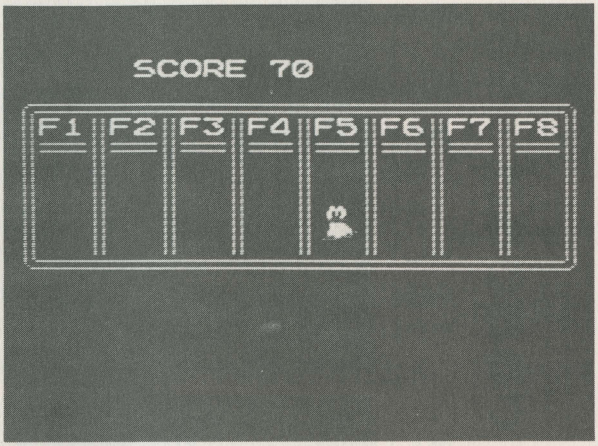
真空管にしろトランジスタにしろ，熱には弱く，人間のいるオフィスにはクーラーなんかなくても，コンピューターを置いてある部屋は完全な空調設備を完備していた。コンピュータールームでは勿論，タバコなんかは絶対禁止，軍隊や，最新の機械は，厳重な身分チェックを行ったあとではじめてコンピューターにさわれたんだ。

今，16ビットという，テレビ・本体・プリンターを全部ふくめて 100万円位のコンピューター，10数年前ならいくらすると思う。何千万円から何億円もしたんだ。それくらいコンピューターは安くなり，高性能になってるんだ。

10年後は，いったい，どうなるんだろうね！

ペンギンたたき





オレは，アメリカのNASAの宇宙飛行士だ。今や，スペースシャトルに乗ることは，あたり前のこととなった。今回のオレの飛行は，地球のおえらいさんを，「静の海」に降ろし，拾うことだ。何でも，先頃第三種接近遭遇した，最初の異星人との重要な会合を，人類の宇宙への1歩が印した記念の月面上ドームで開こうということらしい。

今，お客をアームストロング記念大ホール（よく解らんが，人間で最初に月に降りたヤツの名前らしい）に届け，離陸して来た所だ。今から1週間，月の上空で待っていて，客を地球に送り帰さねばならない。1週間たった1人ぽっちとは，…。

これも，スペースシャトルがハイヤーがわりに使われるようになったためで，人員削減とか言って，1人でシャトルを動かすはめになってしまったのだ。その代わり，コンピューターは5台もある。信頼性を上げるためだか何だか知らぬが，冷蔵庫まで8個も置くことは，ムダのような気もする。

まあ，ごたくはこのぐらいにして，1週間，羽を伸ばすとしますか。

―――――――――，？―――!!？

冷蔵庫を開けたオレは，思わず声をあげそうになった。ペ，ペンギン!?と見る間にその太ったペンギンが，消えてしまうではないか。これがエイリアンか？。そういえば，なにかの鳥に似てるとか言っていたな，と頭の中で思いながら，目はオレの大切な食い物の方へ吸い

プログラムリスト

```
10 SPRITEOFF:CGSET1,0
20 DIMA(23):B$="BONUS HIT":K1=10:LOCATE10,1:PRINT"
モチテン";K1
100 CLS:LOCATE10,1:PRINT"モチテン";K1:FORI=0TO23:A(I)=
0:NEXT:A1=0:T1=0:F1=0:F2=0:F3=0:F4=0:I1=0:GOSUB100
0:GOSUB2000:X=0:Y=3:GOSUB1100:T1=SU:A1=AA
110 A2=0:T2=0:GOSUB1000:X=0:Y=13:GOSUB1100:T2=SU:A
2=AA:Y=3
200 LOCATE6,12:INPUT"カケテン";K:IFK<1ORK>K1 THEN200
210 GOSUB1000:GOSUB2000:X=X+3:IFX=27THENX=0:Y=8
220 GOSUB1100:T1=T1+SU:A1=A1+AA
225 IFT1>21THEN260
230 IFSTRIG(0)=4THEN210
240 IFSTRIG(0)=8ANDT1>11THEN300
250 GOTO230
260 IFA1=0THEN300
270 A1=A1-1:T1=T1-10:GOTO225
300 X=0:Y=13
310 GOSUB1000:X=X+3:IFX=27THENX=0:Y=18
320 GOSUB1100:T2=T2+SU:A2=A2+AA
330 IFT2<16THEN310
340 IFT2>21THEN370
350 IFT1<22ANDT2<T1 THEN310
360 GOTO400
370 IFA2=0THEN400
380 A2=A2-1:T2=T2-10:GOTO330
400 GOSUB600:LOCATE6,12:IFF1=11ANDF4=2THEN430
410 IFT1>21THEN415
```

よせられていった。
食・わ・れ・て・る・!
1週間も食わずにいれば，死んでしまう。
頭に来たオレは，地球のお偉い方のことなどかまわずに，このペンギンもどきのエイリアンを追い出すことにした。
ヤツらが出て来た所で，ひったたくと，どこかに消えてしまうが，うしろ向きの時にたたくと，冷蔵庫の奥へもぐり込んでしまう。
次から次へと現われるペンギンをたたきながら，目の前が暗くなって行くのを感じていた。いったい，何匹いるんだー。

★遊び方

プログラムとBGグラフィックを打ち込んで，RUNすると，ゲームが始まります。
画面の8つの冷蔵庫の中に，ペンギンが次々と，現われては消えていきます。ペンギンが画面中に出ている間に，それぞれに対応するファンクシャンキーを押すと，ペンギンをたたいたことになります。
前向きのペンギンをたたくと＋1点加算され，後ろ向きのペンギンをたたくと，1点減点されます。一定時間内に何点取るかを競います。
点数は常に画面中に表示されています。

★プログラムの説明

プログラムは，見ていただければわかるとおり，いつ，どこにペンギンが出るかは，RNDで決めてありますので，みなさんが好みの数に変えて，ゲームの難易度をいじくって見てください。ペンギンのキャラクターがいやな人は，他のキャラに変えてください。これは，他のゲームでも同じことです。

★ペンギンたたきについて

モグラたたきゲームを覚えていますか。これはあのモグラたたきと同じです。
基本的に，反射神経に左右されるようなゲームでは，ハイパーオリンピックや早射ちゲーム，モグラたたき，などがありますが，モグラたたきに，ただ反射神経にだけたよるわけではない早射ちゲームの「おてつき」の考えを入れたものが，このペンギンたたきゲームです。
この種のゲームを（まあ，何のゲームでも当てはまることですが）作る際に，気を付けることは，ゲームのテンポです。このプログラムのPAVSE文の数値を10倍ぐらいにしてごらんなさい。確かにスリリングにはなるかも知れませんが，ゲームを行なって「燃える」よう

```
411 IFT2>21THEN440
412 IFT1>T2 THEN440
413 GOTO420
415 IFT2>21THEN480
420 PRINT"COM WIN":K=-1*K:GOTO500
430 FORI=0TO3:GOSUB600:PAUSE40:LOCATE6,12:PRINT"BL
ACK JACK";:PAUSE40:NEXT:K=K*10:GOTO500
440 IFF3=3ANDF4=3THENPRINTB$:K=K*7:GOTO500
450 IFF2=111ANDF4=3THENPRINTB$:K=K*5:GOTO500
460 IFF4>4THENPRINTB$:K=K*3:GOTO500
470 PRINT"YOU WIN":GOTO500
480 PRINT"DROW":GOTO510
500 K1=K1+K:LOCATE14,1:PRINTK1
505 IFK1<1THENEND
510 IFSTRIG(0)<>4THEN510
520 GOTO100
600 LOCATE6,12:PRINT"              ":RETURN
1000 GA=RND(4):SU=RND(13)+1:I=0:DM=GA*13+SU
1010 IFI=I1 THENA(I1)=DM:RETURN
1020 IFA(I)=DM THEN1000
1030 I=I+1:GTO1010
1100 AA=0:IFGA=0THENG$="S"
1110 IFGA=1THENG$="H"
1120 IFGA=2THENG$="K"
1130 IFGA=3THENG$="D"
1140 IFSU<10THENS$=MID$(STR$(SU),2,1)
1150 IFSU=10THENS$="T"
1160 IFSU=11THENS$="J"
1170 IFSU=12THENS$="Q"
1180 IFSU=13THENS$="K"
1183 IFSU>10THENSU=10
```

なことはないでしょう（これも好みかも知れませんが)。また，誰もが不可能なほど，速いゲームも，実際プレイできません。

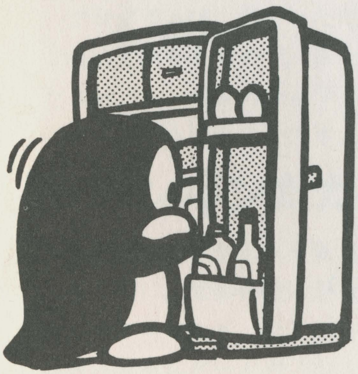
ゲームの作成上，もう１つ考えなければいけないことに「ゲーム・バランス」があります。あまりに簡単なゲームもあまりに難かしいゲームも，ゲームバランスからすると，良いゲームとは言えません。ゲームバランスを良くする方法として，プレイヤーが上達して行くにつれ，難かしくすることが考えられます。

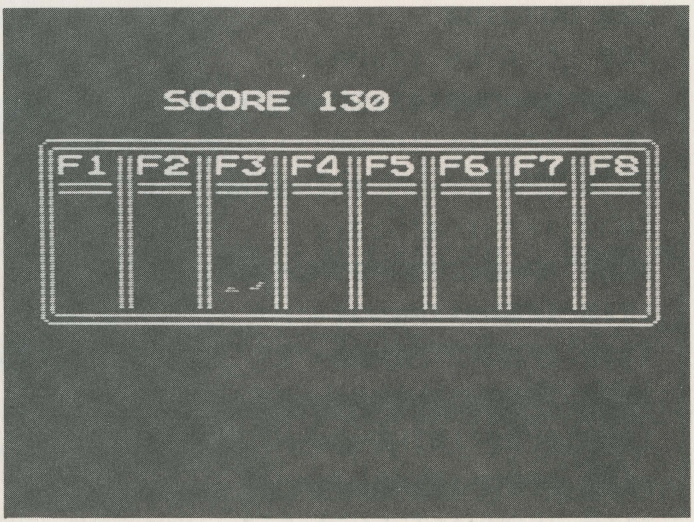
このやりかただと，つねにプレイヤーの腕に合ったレベルのゲームが存在することになりますので，「１回やったら，もうそれっきり」ということは，起こりにくくなります。

このような反射神経ゲームでは，忍耐力と集中力が物を言いますので，難易度の傾斜をつける必要はありませんが，他のリアルタイムゲーム，知能ゲームなどを作る時には，ぜひ考えて欲しいものです。

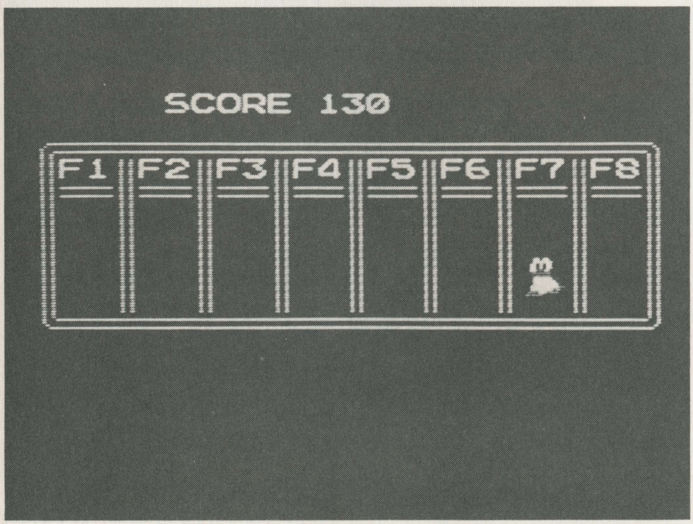
ゲームは想像力なり——古代クレタ人のことわざ。—なんて，……。

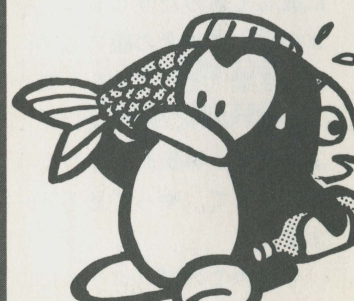
プログラムは，無限の広がりがあります。苦労すればするほど，面白いものが出来上がります。がんばってください。



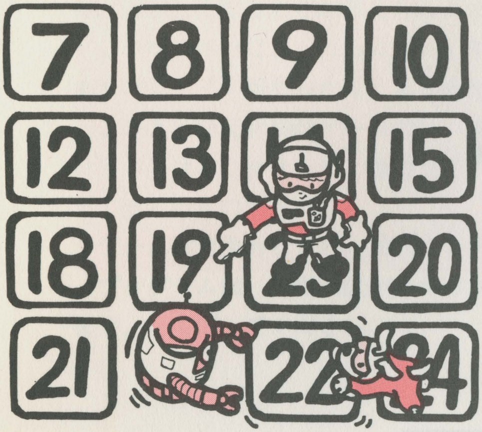


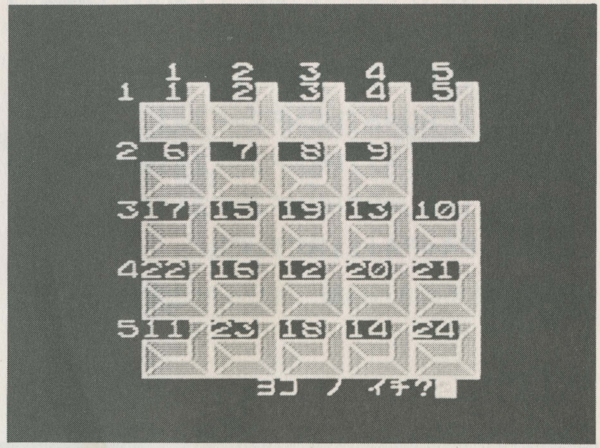
```
1185 IFSU=1THENS$="A":AA=1:SU=11
1190 LOCATEX,Y:PRINTCHR$(222);CHR$(227);CHR$(223);
:LOCATEX,Y+1:PRINTCHR$(226);G$;CHR$(226);:LOCATEX,
Y+2:PRINTCHR$(226);S$;CHR$(226);:LOCATEX,Y+3:PRINT
CHR$(224);CHR$(227);CHR$(225);:RETURN
2000 IFDM=1THENF1=F1+1
2010 IFDM=11THENF1=F1+10
2020 IFSU=6THENF2=F2+1
2030 IFSU=7THENF2=F2+10:F3=F3+1
2040 IFSU=8THENF2=F2+100
2050 F4=F4+1:RETURN
```





5 スクウェア





一枚の板の上に碁盤の目状に正方形のコマが並んでいる。そのコマの１つが取れ，空いた部分に，外のコマが移動出来る。

こんなゲーム（パズルかな？）見たことあるでしょう。旅行とか遠足とかになると，友人の１人ぐらいは，必ずこのたぐいのものを持って来ていたり，自分でも１つぐらいは家の中に放ってあったりします。玩具屋の中を見渡すと，色々なバリエーションがあります。３×３のビーグル犬の絵のもの，４×４の旅行用のもの等々。中には誰がやるのか，18×12など と大きな（そしてかなり難かしい）ものまであって，興味をそそります。

このパズル，欧米人にはごく当り前の良く知られたもので，かなり古くから存在しているようです。事務計算を行なうようなコンピューターにも，息抜き用にこのパズルが乗っていたりもして，やって見ると，疲れた時のコーヒーブレイクにはもってこいのパズルだと解ります。

今回のこのプログラムは，５×５の最もスタンダードなもので，応用も色々と可能なものです。

★遊び方

プログラムをRUNすると，１から24までの番号が描いてあるコマが，でたらめに並んで

プログラムリスト

```
10 CLS
100 DIM MP(4,4),RN(24):E$=CHR$(&HFD)+CHR$(&HFD)+CH
R$(&HFD)
110 OX=5:OY=5
200 FORA=0TO24
210 IX=A MOD 5:IY=A/5
225 MP(IY,IX)=A+1:NEXT
230 FORX=0TO4:LOCATE3*X+OX,OY-1:PRINTX+1;:NEXT
235 FORY=0TO4:LOCATEOX-2,3*Y+OY:PRINTY+1;:NEXT
240 FORX=0TO4:FORY=0TO4:IFMP(X,Y)=25THEN260
250 D=MP(X,Y):GOSUB3100
260 NEXT:NEXT:X=4:Y=4
300 GOSUB1000:GOSUB3000:GOSUB4000
310 IFEN=0THEN300
320 LOCATE10,20:PRINT"ｾｲｺｳ !!";:PAUSE10:LOCATE10,2
0:INPUT"ﾓｳｲﾁﾄﾞ ﾔﾘﾏｽｶ Y/N";A$
330 IF A$="Y"THENRUN
340 END
1000 LOCATE10,20:INPUT"ﾖｺ ﾉ ｲﾁ";PX:LOCATE10,20:INP
UT"ﾀﾃ ﾉ ｲﾁ";PY:PX=PX-1:PY=PY-1:IF(PX>4)OR(PX<0)THE
N1000
1010 IF(PY>4)OR(PY<0)THEN1000
1020 LOCATE10,20:PRINT"            ";:RETURN
3000 VX=SGN(X-PX):VY=SGN(Y-PY)
3010 IF(VX<>0)AND(VY<>0)THEN3099
3020 IF(VX=0)AND(VY=0)THEN3099
3030 IFVY<>0THEN 3055
3040 DY=Y:FOR DX=X TO PX STEP SGN(PX-X)
```

出て来ます。これを左上が1，左下が21になるように並べ換えると，ゲーム終了です。コマはマス目の中に1つだけ存在する“空き目”にのみ移動可能です。

ただし，実際にコマを動かす時には，“空き目”までの複数個のコマを同時に動かすことが出来ます。つまり，“空き目”の上下左右に並んだいくつかのコマは，一動作で“空き目”を埋めるように一つずつ，ずらすことが可能なのです。コマの座標を入力すると，そのコマが動ければ，指定したコマから“空き目”までが1つずつ，“空き目”の方に移動します。こうして1から24までのコマが，指定された場所に入れると，GAME OVERとなります。

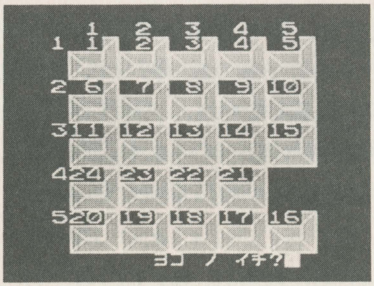
このプログラムには，左上から右下へ並べるような，終了判断のルーチンをつけていますが，このルーチンを使わずに，並べ方を変えると，それだけで難しさは変化します。右下から左上へ，上から下へ，1つとびになど，色々と試してください。

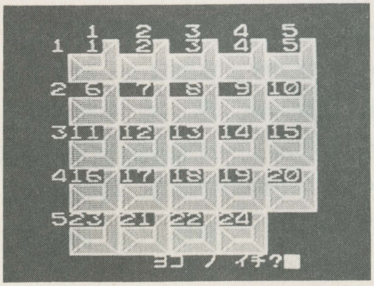
★プログラムの説明

F8 KEYでプログラムを走らせます。最初のコマの並び方を見てください。右上と左下がひっくり返っているはずですね。ちょっと見ると，これは楽に出来るかな？と思うかもしれませんが，やってみるとけっこう大変です。動かすコマの位置をKEYで入力するのですが，2，3個まとめて動かすことも出来ます。やってみてください。たとえば，右下すみに穴がある時に，ヨコ5タテ1をKEY INすると（5－2）(5－3)(5－4)の3つも同時に下に1個ずれるのです。

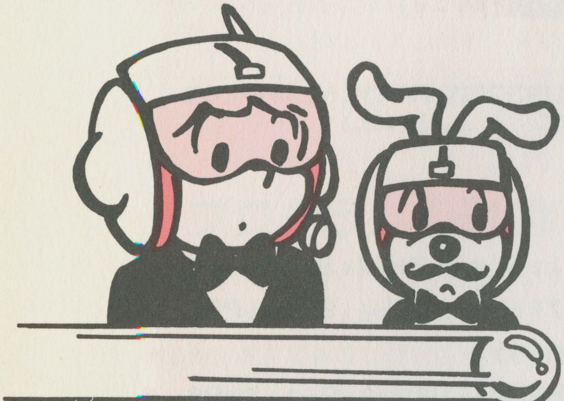
（5－2）を指定すれば（5－3）と（5－4）も同時に動きます。数字のKEYの後にリターンキーを押して始めて入力します。プログラムの中をのぞいてみましょう。X，Yはコマのない位置を示します。PX，PYはプレイヤーが指した位置，VX，VYはコマを動かす方向です。グラフィックの位置は左上が（0－0）ですから，右に1つ動くということは，ヨコの位置に1を加えることになります。ここではVXが1，VYが0です。VYはタテの動きですから，動かないということは0を足すことになるのです。コンピューターはPX,PYにあるコマをX，Yに移すことをくり返しているわけです。すべてのコマのデータはMP(4,4）の変数に入っています。配列変数MP（ ， ）はMP（横の位置，縦の位置）で，グラフィック上の位置より1少ない値数，つまり，MP（0，0）はグラフィックでは位置（1－1）を示しています。穴のある位置のデータは25にしてありますから，例えば（1－1）と（1－2）を入れ換える時にはMP（0，0）にMP（0，1）に入っていた数を入れ，MP(0，1）にはMP(0,0）に入っていた数を入れるのです。このように，位置と変数をむすぶ使い方を，変数にマップする（マッピングする）と言います。

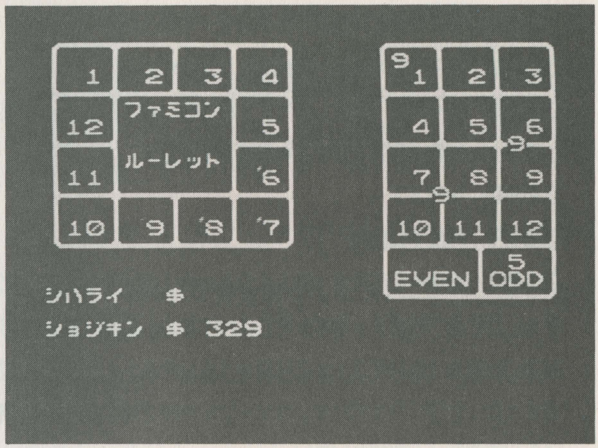
```
3045 IF DX=X THEN 3050
3047 D=MP(DX,DY):GOSUB3100:GOSUB3200:MP(X,Y)=MP(DX
,DY):MP(DX,DY)=25:X=DX
3050 NEXT
3055 IFVX<>0THEN3099
3060 DX=X:FOR DY=Y TO PY STEP SGN(PY-Y)
3070 IF DY=Y THEN 3090
3080 D=MP(DX,DY):GOSUB3100:GOSUB3200:MP(X,Y)=MP(DX
,DY):MP(DX,DY)=25:Y=DY
3090 NEXT
3099 RETURN
3100 D$=STR$(D):D$=RIGHT$(D$,2):SX=X*3+OX:SY=3*Y+O
Y:LOCATESX,SY:PRINTD$;CHR$(&HF7):LOCATESX,SY+1:PRI
NTCHR$(&HF3);CHR$(&HF6);CHR$(&HF4);
3110 LOCATESX,SY+2:PRINTCHR$(&HF5);CHR$(&HF6);CHR$
(&HF8);:RETURN
3200 SX=DX*3+OX:SY=DY*3+OY
3210 LOCATESX,SY:PRINT"   ";:LOCATESX,SY+1:PRINT"
   ";:LOCATESX,SY+2:PRINT"   ";:RETURN
4000 EN=1:FORA=0TO24:IFMP(AMOD5,A/5)<>(A+1)THENEN=
0
4010 NEXT:RETURN
```





ルーレット





ルーレットはみんな知ってるでしょう？　クルクルまわるあれです。本場のカジノでもイチバン花やかなのがルーレットです。ほかのカードゲームなんかが割と小じんまりとやっているのに比べて，中央にドーンと広がっている感じがします。そのまわりには立派な身なりの人たちが余裕タップリで遊んでいる感じ（あれは絶対に農協の団体旅行では似あわない!!）。

どうせ日本では経験できないし，わざわざ外国まで行けるわけでもないので，ファミコンを使って練習しておいて，もし実際にやる機会があったら大もうけしましょう。その時はディラーさんにチップをあげるのをお忘れなく。

★ファミコン版ルーレットの説明

念のためルーレットの説明を簡単にしておきますと，要するに次に出る数字を予想して当てればいいわけです。本物は１～36までと，０，００の計38個の数があって，数字以外の賭け方も奇数，偶数，赤，黒，大，小などなどたくさんあるんですが，ファミコン版はいろいろな制約上，数字は１～12まで，その他は奇数，偶数だけになっています。

予想が当たった時の倍率は，賭け方によって次のようにわかれています。

① **１点賭…………12倍**

② **２点賭…………６倍**

プログラムリスト

```
100 CLS:SPRITEON:CGEN 3
110 CGSET1,0:PALETB 2,47,48,39,48
120 DEFSPRITE0,(3,0,0,0,0)=CHR$(207)
130 DIM X(11),Y(11),B(64)
140 FORI=0TO3:X(I)=I*24+36:Y(I)=44:X(I+6)=108-I*24
:Y(I+6)=116:X(I+3)=108:Y(I+3)=I*24+44:NEXT
150 FORI=0TO2:X(I+9)=36:Y(I+9)=116-I*24:NEXT
160 M=20
500 IFM=0ORM>30000THEN4000
510 FOR I=8 TO 64:B(I)=0:NEXT
520 VIEW:GOSUB4500
530 GOSUB1000:GOSUB2000:GOSUB3000:GOTO500
1000 X=172:Y=44:I=8:SPRITE0,X,Y
1010 GOSUB1800:IFK$=CHR$(13)THENRETURN
1020 IFK$="E"THENX=184:Y=140:I=63:GOTO1200
1030 IFK$="O"THENX=220:Y=140:I=64:GOTO1200
1040 IFX>172ANDK$=CHR$(29)THENX=X-12:I=I-1
1050 IFX<220ANDK$=CHR$(28)THENX=X+12:I=I+1
1060 IFY>44ANDK$=CHR$(30)THENY=Y-12:I=I-7
1070 IFY<116ANDK$=CHR$(31)THENY=Y+12:I=I+7
1080 IFASC(K$)>48ANDASC(K$)<58GOSUB1500
1090 SPRITE0,X,Y
1100 GOTO1010
1200 SPRITE0,X,Y:GOSUB1800
1210 IFASC(K$)<49ORASC(K$)>57THEN1000
1220 GOSUB1500:GOTO1000
1500 J=VAL(K$):IFJ>M OR M=0 THEN RETURN
1510 IF B(I)>0 THEN RETURN
```

③　4 点賭…………3 倍

④　奇数，偶数……2 倍

2 点賭や 4 点賭というのは 1 枚のチップを 2 つ，または 4 つの数字にまたがって賭けることです。この時はチップを数字の境界の上に置きます。

★遊び方

最初は所持金＄20でスタートです。

ゲームが始まると，右側の 1 ～12と EVEN（偶数），ODD（奇数）の中の 1 の所に赤丸が表示されます。これがカーソルで，カーソルキーで上下左右に動かせます。ただしカーソルキーで動かせるのは数字の部分だけで，EVEN，ODD にカーソルを持っていく時は，“E”，“O”をそれぞれ押してください。

賭け金は 1 ヶ所に＄1 ～＄9 までで，賭けた金額が表示されます。所持金は画面の左下に表示されます。

賭け終わったら RETURN キーを押してください。左の数字の上を赤丸がクルクルまわり，しばらくすると止まります。

当たればその分が払い戻されて，画面の下に，

HIT ANY KEY !!

と表示されますから，好きなキーをひとつ押してください。所持金が 0 になるか，＄30,000 を越えればゲーム終了です。

★プログラムの説明

プログラムは，イニシャライズ，数字を予想して賭ける，ルーレットの回転，支払の 4 つにに大きく分けられます。

●イニシャライズ　（100～160）

100～120　画面初期設定，色指定，スプライトの定義

130　配列宣言

140～150　ルーレットの球の表示位置データセット

160　所持金初期値

●数字を選んで賭ける（1000～1810）

1020～1070　カーソル移動

1090　カーソル表示

1200～1220　偶数，奇数に賭ける

1500～1530　賭け金セット

```
1520 M=M-J:B(I)=J:GOSUB4500
1530 LOCATE(X-16)/8,(Y-24)/8:PRINTHEX$(J):RETURN
1800 K$=INKEY$:IFK$=""GOTO1800
1810 FORK=0TO199:NEXT:RETURN
2000 A=RND(12)
2010 FORW=1TORND(2)+1
2020 FORJ=0TORND(3)+1
2030 FORI=0TO11:SPRITE0,X(I),Y(I):PLAY"B0"
2040 FORK=0TOW*20:NEXT:NEXT
2050 NEXT:NEXT
2060 FORI=0TOA
2070 SPRITE0,X(I),Y(I):PLAY"B0"
2080 FORJ=0TO(W+I)*40:NEXT:NEXT
2090 RETURN
3000 I=A/3*14+(A MOD 3)*2+8
3010 J=B(I)*12
3020 J=J+(B(I-7)+B(I+7)+B(I-1)+B(I+1))*6
3030 J=J+(B(I-8)+B(I-6)+B(I+6)+B(I+8))*3
3040 IF A MOD 2=0 THEN J=J+B(64)*2:GOTO3060
3050 J=J+B(63)*2
3060 IFJ>0GOTO3080
3070 PLAY"CDCE":GOTO3120
3080 PLAY"BAGE"
3090 FOR I=J TO 1 STEP-1
3100 LOCATE8,16:PRINTJ;" ":PLAY"E0":M=M+1:GOSUB450
0:J=J-1
3110 NEXT:LOCATE8,16:PRINT0
3120 LOCATE 5,21:PRINT"HIT ANY KEY!!";:PAUSE
3130 RETURN
4000 VIEW:LOCATE 10,21:PRINT"GAME OVER"
4010 LOCATE 5,22:PRINT"HIT ANY KEY";:PAUSE
```

1800～1810　キー入力ルーチン

●ルーレット回転　（2000～2090）

2000　数字を乱数で決める

2040～2040　タマを回転させる

2060～2080　決めた数字にタマを止める

●支払い　（3000～　）

3010　１点賭けチェック

3020　２点賭けチェック

3030　４点賭けチェック

3040～3050　偶数，奇数賭けチェック

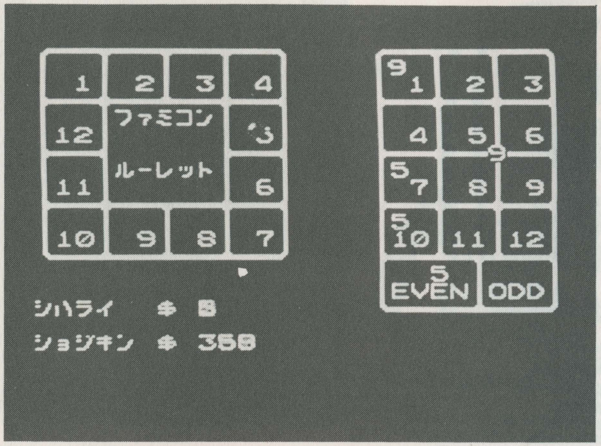
3090～3110　支払い表示

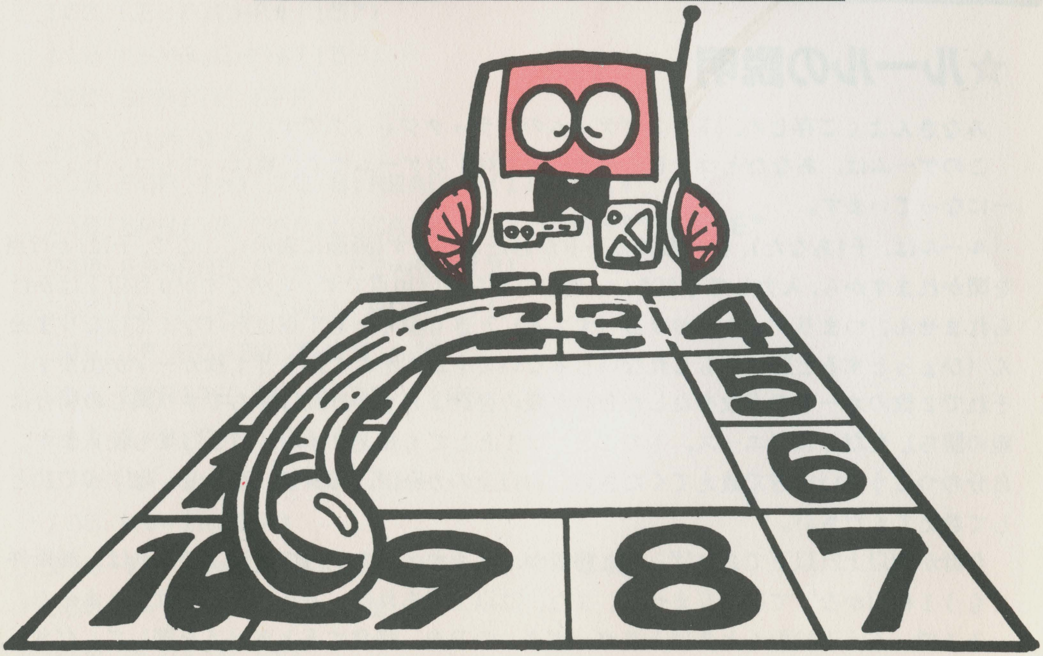
3120　キー入力待ち

★変数表

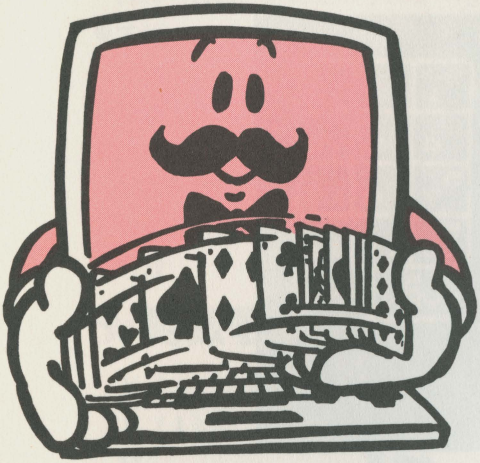
X（I），Y（I）	ルーレットのタマ表示位置
B（I）	賭け金と場所保存用
M	所持金
X，Y	カーソル表示位置
W	タマの回転速度制御用
K＄	キー入力用
I，J，K	汎用変数

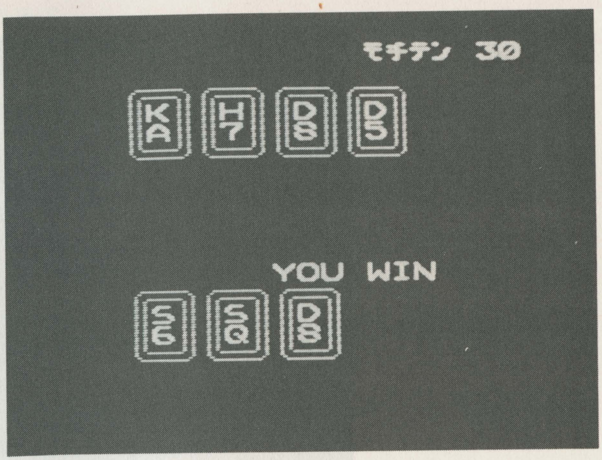
```
4020 CLS:SPRITE OFF
4030 END
4500 LOCATE8,18:PRINTM;"    "
4510 RETURN
```





ブラックジャック





★ルールの説明

みなさんよくご存じの，トランプゲームのブラックジャックです。

このゲームは，あなたとコンピューターの１対１のゲームです。親はいつもコンピューターになっています。

ルールは，子(あなた)，親の順にカードがわたされます(画面に表示)。ここで，子は，かけ点を聞かれますから，入力してください。最初のもち点は10点です。当然ですが０点以下はかけられません。つまりゲームをおりる（すてる）ときも少くとも１点はかけなくてはなりません（ひょっとすると勝つかもしれないじゃないですか)。そこで，もう１枚カードが出ます。それで２枚のカードの点数をたした合計が親の合計よりも多ければ勝ちです（同じの場合は親の勝ち)。ただしA（エース，１のこと）は１としても使えますし，11としても使えます。自分のつごうのいい方で数えてください。10以上のカード（10，J，Q，K）はすべて10として数えてください。

合計が12以上21以下であれば，親と勝負ができます。したがって11以下であれば，無条件でもう１枚引かなくてはなりません。また，22以上であれば，もう引くことはできません。

そこで，もう１枚引くときはトリガーボタンのBを，勝負するときはAを押してください。

プログラムリスト

```
10 SPRITE ON:CGSET 1,0
20 DEF SPRITE 0,(0,1,0,0,0)=CHR$(104)+CHR$(105)+CH
R$(106)+CHR$(107)
30 DEF SPRITE 1,(0,1,0,0,0)=CHR$(108)+CHR$(109)+CH
R$(110)+CHR$(111)
35 VIEW
40 KEY1,CHR$(151)
50 KEY2,CHR$(152)
60 KEY3,CHR$(153)
70 KEY4,CHR$(154)
80 KEY5,CHR$(155)
90 KEY6,CHR$(156)
100 KEY7,CHR$(157)
110 KEY8,CHR$(158)
200 SPRITE OFF
210 DIM A(8)
220 FOR I=1 TO 8:READ A(I):NEXT
230 SPRITE OFF:LOCATE 6,3:PRINT"SCORE"
240 PS=RND(8)+1:P2=RND(2)
245 IF P2=0 THEN PC=PC+1
246 IF PC=30 THEN 600
250 SPRITE P2,A(PS),105
260 I=0:SPRITE ON
270 IF I=35 THEN 310
280 K$=INKEY$
290 IF K$=CHR$(150+PS) THEN 340
300 I=I+1:GOTO 270
310 SPRITE OFF
```

こんどは親の番です。親は，16以上21以下でないと，勝負できません。したがって，いつもコンピューターが親では不公平だと思われた方も安心してください（実際にけっこう勝てますよ）。そして親が何枚か引いて，手ごろな合計になったときに，勝負です。前にも言いましたが，勝つためには，21以下で，なおかつ相手より大きい数であればいいのですが，同点のときは親の勝ちです。これで，1回の勝負は終わりです。トリガーボタンAを押すと，次の勝負スタートです。

ところで，子には，あるきまりどおりに，カードをそろえると，BONUS POINTとして，かけ点の数倍の点数が返ってくるようになっています。BONUSとしては，まず5枚以上のカードを引いて勝った場合は，かけ点の3倍が返ってきます。次に6，7，8で21になり勝った場合はかけ点の5倍が返ってきます。次に7，7，7で21になり勝った場合はかけ点の7倍が返ってきます。最後は，BLACK JACKです。これはスペードのJ（S—J）とスペードのA（S—A）2枚のみで勝負をしたときは，たとえ親が21でも子の勝ちです。つまり一番強い無敵の手です。これはかけ点の10倍が返ってきます。ただし，親が今まで書いたBONUSを出しても，何倍も取られることはありません（あしからず）。

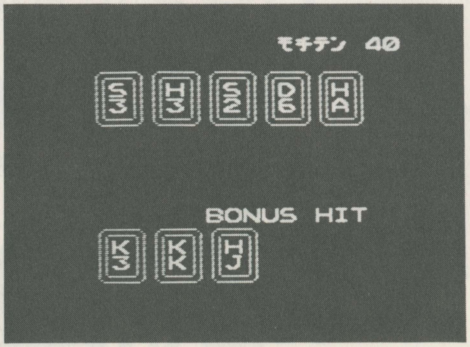
さらに，カードの見方として四角で囲まれた中に，上にSとかKとか書かれますが，上は，Sはスペード，Hはハート，Kはクラブ，Dはダイアを意味し，下の数字やAとかKは2か9までは，そのまま2から9で，Aは前に書いたように，エースを意味し，1または11です。あとTは10（英語のTENからとりました。10だと，表示できませんので。）Jは11，Qは12，Kは13です。もちろんTからKは10から13なのですが，このゲームでは，すべて10とします。（つまり，スペードのエースと，スペードのジャック—J以外は，あまり意味はないのですが……）

★プログラムの説明

プログラムについては，**100行**からがメインルーチンです。自分のカードとコンピューターのカードを表示したあと**200行**でかけ点を聞いてきます。**201行から270行**までは，自分のカードを引くルーチンになっています。21をこえると，無条件でコンピューターのカードを引くルーチンに行きます。12より少ないなら，勝負のキー，トリガーボタンBが押せなくなっています。この点を変えれば，12以上になるまで勝手にカードを引くようにもできます（自分で考えてみてください）。**300行から380行**までがコンピューターのカードを引くルーチンです。ルールどおりに，16以上になるまでカードを引き，21以下で，自分が21以下でコンピューターより大きいときのみ，もう一度カードを引くようになっています。**400行から480行**までは，どちらが勝ったかをしらべて表示し，BONUSのときは，その処理をします。

500行から520行は，もち点が0になったかどうかをしらべ，そうでないなら再びゲームを

```
320 PAUSE RND(500)
330 SPRITE 0:SPRITE 1:GOTO 240
340 CGSET 1,2:BEEP
350 IF P2=1 THEN SC=SC-10:GOTO 370
360 SC=SC+10
370 CGSET 1,0
380 LOCATE 6,3:PRINT"SCORE";SC:GOTO 310
600 LOCATE 3,20:PRINT"REPLAY ? Y OR N"
610 IF INKEY$<>"" THEN 610
620 IF INKEY$="Y" THEN RUN
630 IF INKEY$="N" THEN 700
640 GOTO 620
700 KEY1,"LOAD"+CHR$(13)
710 KEY2,"PRINT "
720 KEY3,"GOTO "
730 KEY4,"CHR$("
740 KEY5,"SPRITE"
750 KEY6,"CONT"+CHR$(13)
760 KEY7,"LIST"+CHR$(13)
770 KEY8,"RUN"+CHR$(13)
1000 DATA 33,56,80,104,128,152,176,200
```





続けます。

600行は，BONUSのときや，その他のときの表示を消すルーチンです。

1000から1030行までは，カードを引いてくるルーチンです。ただし，同じカードを引かないように，しらべでいます。そのための記憶に**20行**で DIM　A（23）として変数Aを使っています。変数の内容は，スペードが1から13，ハートが13＋1から13＋13まで，あとはクラブ，ダイアというふうに52までの数を使っています。

1110行から**1190行**は，**1000行**で出した数から画面に出す表示を決め，表示しているルーチンです。数値変数というのは，表示するとき十，一の区別のため，正の数には必ずスペースが1つ入ってしまうので，まずそれを取らなくてはなりません。それが，**1140行**で行なわれているのです。これは一種のプログラミングの上でのテクニックですので，覚えていると便利でしょう。

そして，**2000行**からは，BONUSに必要なスペードのエース，スペードのジャック，6，7，8や，引いた枚数などを数えるルーチンです。これだと，BONUSの6，7，8は，スペードとか，クラブなど，かまわず6，　7，　8とあればBONUSになるようにしてありますが，ここを改良すると，スペードで6，　7，　8としなければ，BONUSにならないようにすることも可能でしょう。

なにぶん2キロのベーシックでも動くようにしたため，BONUSの種類もあまり多くできないし，多人数でもできるものは作れませんでした。4キロベーシックをお持ちの方は，いろいろと手を加えてみるのもいいでしょう（ただし，画面の表示には気をつけてください。あまり多くは表示できませんから)。

ゲームをする上でのテクニックとしては，自分の合計は少なくても，コンピューター側に21より大きい数を引かせて，勝つという方法もあります。21に近い数を出すことだけが，勝ちにつながるとは限りません。21を越えたら，良くて引き分けですから，よく考えてやってみてください。

それでは，健闘を祈ります。



ロード・ランナー

「ロードランナー」というのはファミコンや数多くのパソコンに移殖された アメリカの大ヒットゲームだね。でも，もうひとつの意味があるって知ってるかな。コン通（コンピューター通）の君なら知ってるね。

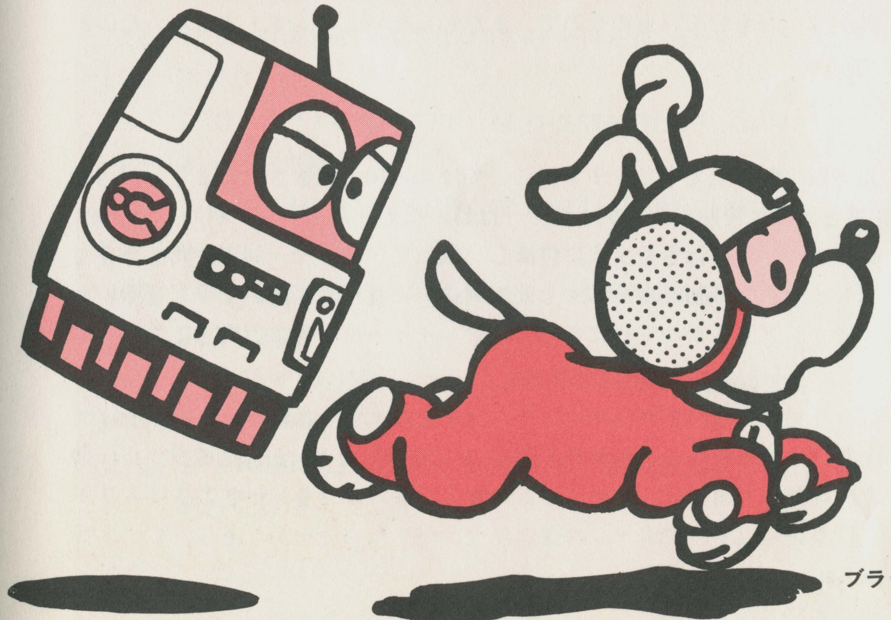
普通のパソコンはプログラムはテープやディスケットに入っている。そのプログラムをコンピューター上に呼び出す命令は「LOAD」（ロード），そして，プログラムを開始する命令が「RUN」(ラン)。つまり，「ロード・ランナー」とは，ベーシックで知っている命令が「LOAD」と「RUN」しかない，というこまった人達だ。

もっとも，ファミコンではプログラムはすべてROMの入ったカセットに入っているし，電源を入れればすぐゲームが始まるから，関係なかったね。でも，ファミリー・ベーシックを手に入れたらそうはいかない。

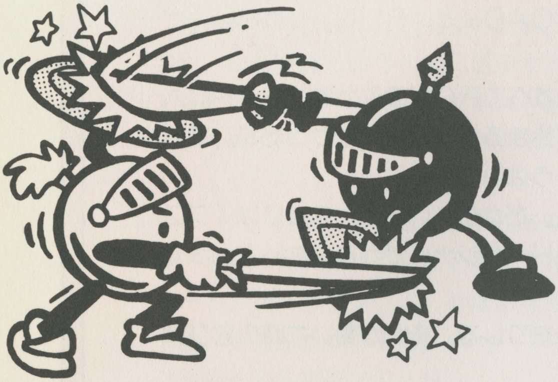
この本を買ってくれた君は「ガンバル人」だからいいけど，なかには「プログラム作りなんてめんどいよ。オレ，ロード・ランナーでかまわないよ」って人がいるかも知れないね。

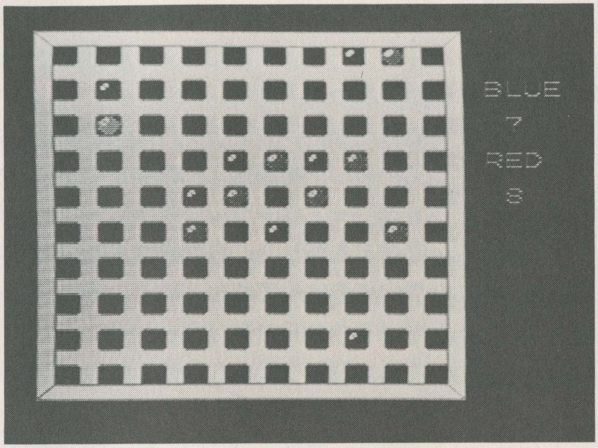
でも，プログラムを入力したり，自作したりして，それが動いた時の喜びは天にものぼる想いだよね！

だから，かれらが「ロード・ランナー」（道を走る人）なら，ぼくらは「スカイ・フライヤー」（空を飛ぶ人）でがんばろう。



ドッジェム





このゲームは，ドッジェムというゲームで，1972年に，アメリカのケンブリッジ大学の学生によって，考えられたものです。今までに，いろいろな本に紹介されて来たものですが，ルールは簡単で，奥が深くおもしろいものなので，まだ知らない人にも覚えてもらいたいと思います。

★ルール

どんな大きさの正方形の盤の上でもできます。ここでは10×10の大きさにしましたが，100×100でも，3×3でも対戦は可能です。プレイヤーは盤の横よりも1少ない数のコマを使い，2人のプレイヤーは，1つの角をはさむように位置し，決してプレイヤー同志が向き合うことはありません。プレイヤーは最初，自分のすぐ前の列に，1番相手に近い1マスを除いたマス目にコマを並べます。どちらかが先行になって，プレイヤーは交互にコマを1つづつ動かします。

各プレイヤーは，自分と向き合う面をゴールにしてコマを進めます。この時プレイヤーはコマを進行方向（つまりプレイヤーから遠ざかる方向），およびプレイヤーから見て右と左に動かせます。プレイヤー自身の方向には，進めません。盤上の敵のコマをどかす方法はありません。敵のコマを避けながら（ドッジボールの様に），コマをプレイヤーから遠ざけていきま

プログラムリスト

```
10 VIEW:CGEN2:CGSET1,0:R=9:B=9:DIMD(9,9),X(17),Y(1
7),Z(17):PALETB 0,13,0,16,2:PALETB 1,13,0,16,22:PA
LETB 2,13,0,16,41
20 FORI=0TO8:D(0,I)=1:D(I+1,9)=2:X(I)=0:Y(I)=I:Z(I
)=1:X(I+9)=I+1:Y(I+9)=9:Z(I+9)=1:NEXT
30 S=0GOSUB40:S=1:GOSUB40:GOTO30
40 GOSUB400:GOSUB100:GOSUB500:RETURN
100 I=S*9:E=0:F=0
110 IFZ(I)=0THENI=I+1:IFI>8+S*9THENI=S*9
120 P=1
130 IFSTRIG(S)=8THENI=I+1:IFI>8+S*9THENI=S*9
150 COLORX(I)*2+2,Y(I)*2+1,2:COLORX(I)*2+2,Y(I)*2+
1,S
160 ST=STICK(S):IFST<>0THEN180
170 GOTO110
180 IFS=0ANDST=1ANDX(I)<9THEN185
182 GOTO190
185 IFD(X(I)+1,Y(I))=0THENGOSUB920:X(I)=X(I)+1:GOS
UB900:RETURN
190 IFS=0ANDST=1ANDX(I)=9THENGOSUB920:Z(I)=0:E=1:R
ETURN
200 IFS=0ANDST=4ANDY(I)<9THEN205
202 GOTO210
205 IFD(X(I),Y(I)+1)=0THENGOSUB920:Y(I)=Y(I)+1:GOS
UB900:RETURN
210 IFS=0ANDST=8ANDY(I)>0THEN215
212 GOTO220
```

す。

一番向こうのコマは，一動作で盤から取り去ることができます。こうして交互にプレイを繰り返していき，早く全部のコマを盤の外に出した方が勝です。

3×3の盤上では，先攻プレイヤーに必勝手がありますが，4×4以上の面では，そのようなものはまだ見つかっていません。10×10の面上の2人ゲームを，誰かと一緒に楽しんでください。

★遊び方

このゲームは2人ゲームですので，コンピューターと対戦することはできません。コンピューターは，プレイヤーの動かすコマの動きをチェックして，勝負の判断を行ないます。プレイヤーは，青と赤との2人です。

青を受け持ったプレイヤーが先攻で，コントローラーⅠを使います。赤のプレイヤーはコントローラーⅡを使ってください。プログラムとBGグラフィックのデーターを入力した後，プログラムを走らせると，青のコマが1つが点滅します。コントローラーⅠのAボタンを押すと，点滅は次のコマに順々に移っていきます。点滅しているコマは，コントローラーのジョイスティックを押すことによって移動させることができます。今一番上にあるコマを右に動かしたければ，Aボタンを押して，一番上のコマが点滅するようにして，右方向のジョイスティックを押します。

青の移動が終われば，次は赤の移動の番です。コントローラーⅡを使って，青の時と同じようにしてコマを動かします。青は右端から脱出し，赤は上の端から脱出します。

画面の右にある数字は，まだ盤上に残っているコマの数で，脱出させるごとに数は減っていき，これが早く0になった方が勝です。相手に勝つためには，なさけは無用です。

敵のじゃまをしながら，いかにして早くゴールまで到達するか。これがゲームの鍵だと思います。

★プログラムの説明

D（9，9）という配列変数を作って面の役割をさせています。この配列の中を，青のコマ＝1，赤のコマ＝2，コマ無し＝0で割りふってあります。X（17），Y（17）の配列はそれぞれのコマのX座標，Y座標を示しています。

10行目からが，イニシャライズルーチンで，ここで最初の設定をすべて行なっています。

100行目からの一番長いルーチンは，動かすコマの選択と，動けるかどうかの判断を行なっています。このロジックをもっと小さいものにすれば，12×12，20×20にも改造可能です。

500行からのルーチンは，勝負判断です。プログラムの速さを求めなければX（17），Y（17）

```
215 IFD(X(I),Y(I)-1)=0THENGOSUB920:Y(I)=Y(I)-1:GOS
UB900:RETURN
220 IFS=1ANDST=8ANDY(I)>0THEN225
222 GOTO230
225 IFD(X(I),Y(I)-1)=0THENGOSUB920:Y(I)=Y(I)-1:GOS
UB910:RETURN
230 IFS=1ANDST=8ANDY(I)=0 THENGOSUB920:Z(I)=0:F=1:
RETURN
240 IFS=1ANDST=1ANDX(I)<9THEN245
242 GOTO250
245 IFD(X(I)+1,Y(I))=0THENGOSUB920:X(I)=X(I)+1:GOS
UB910:RETURN
250 IFS=1ANDST=2ANDX(I)>0THEN255
252 GOTO260
255 IFD(X(I)-1,Y(I))=0THENGOSUB920:X(I)=X(I)-1:GOS
UB910:RETURN
260 GOTO100
400 LOCATE23,5:PRINTR:LOCATE 23,9:PRINTB
410 C=0
420 IF C<9THENQ=0:GOTO425
422 Q=1
425 XX=X(C)*2+2:YY=Y(C)*2+1:IFZ(C)=0THENLOCATEXX,Y
Y:PRINT" ":GOTO440
430 COLORXX,YY,Q:LOCATEXX,YY:PRINTCHR$(207)
440 C=C+1:IFC=18THENRETURN
450 GOTO420
500 IFE=1THENBEEP:R=R-1:IFR=0THENLOCATE23,5:PRINT"
WIN":RETURN700
510 IFF=1THENBEEP:B=B-1:IFB=0THENLOCATE23,9:PRINT"
WIN":RETURN700
520 RETURN
```

の配列は作らなくても良いようにもできます。

★改造へのヒント

このプログラムは，ドッジェム用ですが，判断のロジックを変えるだけで，はさみ将棋，その他のオリジナルゲームの２人用ゲームにできます。その時，勝負の判断のプログラムが，どのぐらい小さくなるかで，コンピューターに乗るかどうかが，決定されます。

簡単な改造としては，

①移動した後，配列Ｄ（　）の中を，０にしないようにする。ただし，Ｘ（17），Ｙ（17）の配列は使わずに，直接Ｄ（　）の配列の中から，座標を拾ってくるようにする。

②相手の２つのコマにはさまれると，コマが消えてしまうようにする。

③盤から脱出するコマは，相手にはさまれたコマだけであるようにする。

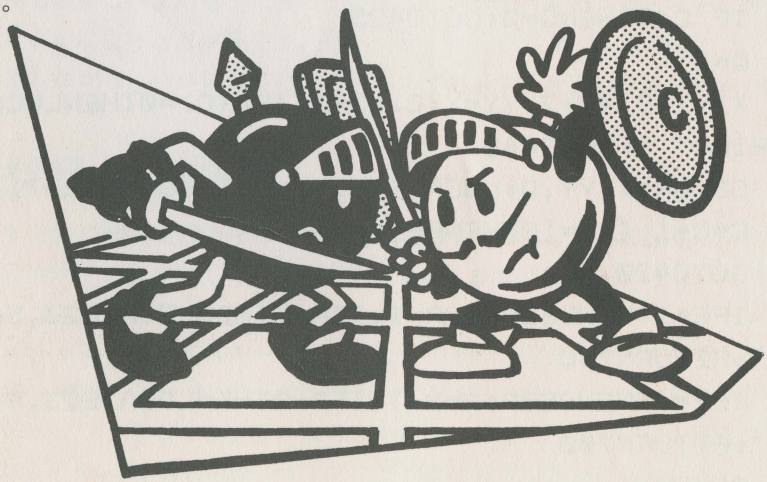
④コマがなくなった方が負けとする。この時，両方とも動けなくなった時はコマの多い方の勝とする。

⑤両方とも，４方向に動けてもよい。これは，プログラムを作る人の好みである。

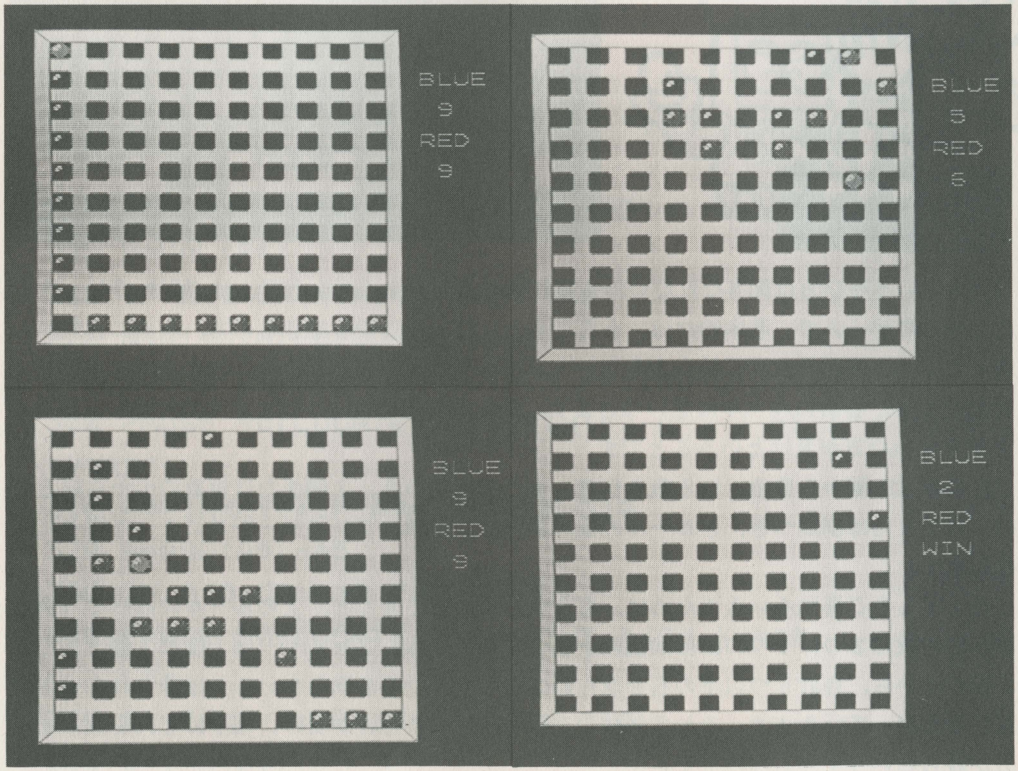
と，以上の５つの事がらを変えてしまえば，このプログラムは，別の思考ゲームとなります。

今，このプログラムは残り十数バイトで走っていますので，もしプログラムを正確に入れたのにＯＭエラーが出て止まってしまう時は，余分なスペース（コマンドと，それに続く文字間のスペース）を取ってみてください。ただし，PALET Ｂの後ろのスペースは必要です。

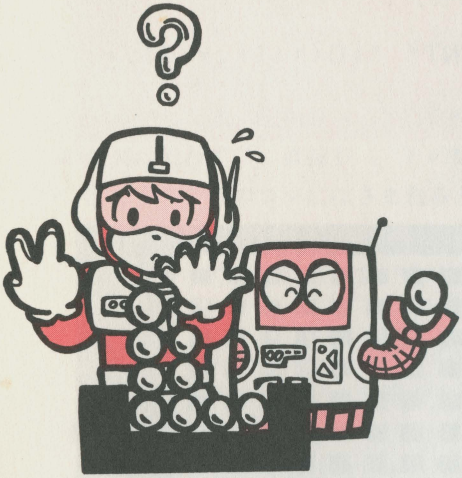
今，１人用のドッジェムも考えていますので，次の機会には，１人でも遊べるようにしたいと思います。

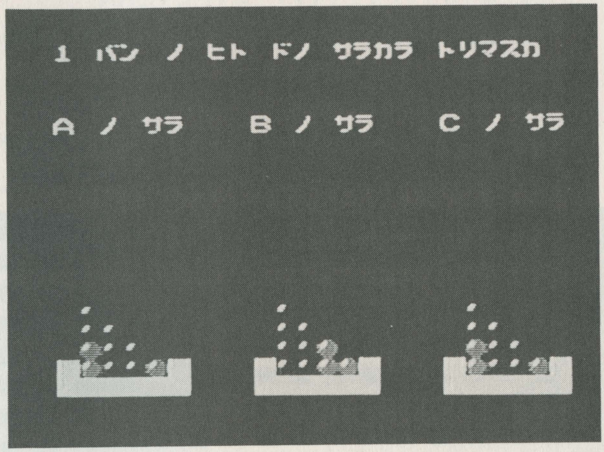


```
700 BEEP
710 IFSTRIG(0)<>0THENRUN
720 GOTO710
900 D(X(I),Y(I))=1:RETURN
910 D(X(I),Y(I))=2:RETURN
920 LOCATEX(I)*2+2,Y(I)*2+1:PRINT" ":D(X(I),Y(I))=
0:RETURN
```



ニム





ニムという呼び方は聞きなれないかも知れませんが，山くずしと言えばご存知でしょう。古くからのゲームで，基本的には，小石でもあれば，どこでもできるので，本当に古代からあったものと思われます。古いが故にバリエーションは数多く存在し，また，ちょっと考えただけでも，容易にルールは思いつきます。最も共通的なルールは，２人で対戦し，最後の１個を取った方が負けということです。

山の数は１個から自由，取る数も任意，または制限つき，１個または複数の山から取るかは，任意または制限つき，とルールの他の部分は，地域，プレイヤーなどによって様々に変化します。

山くずしをご存知ない人のために，もう少しくわしく書くと，上のルールに沿って（例えば，山の数は２つ，１つの山は30個の小石でできている。プレイヤーは１度に１個，または２個の山から，合計５つ以下の石を取らねばならない。石を取らないことは許されない。）交互に石を取っていきます。そして最後の１個の石を取った方が負け，相手にそれを取らせた方が勝ちというゲームなのです。

ルールによって，難易度もかなり変化します。たとえば，一番簡単な例として，山１つ。１個の山は30個の小石でできていて，プレイヤーは３つまで取れるとしましょう。少し考えると，このゲームの必勝法が見つかります。１，５，９，13，17，21，25，29個の小石が，

プログラムリスト

```
100 DIM DX(9),DY(9),N(2),PX(2):VIEW
110 FOR A=0 TO 9:READ DX(A),DY(A):NEXT
115 N(0)=10:N(1)=10:N(2)=10
120 READ PX(0),PX(1),PX(2)
130 GOSUB 1000
160 LOCATE 2,1:PRINT"1 バン ノ ヒト ド ノ サラカラ トリマスカ";
165 A$=INKEY$:IF A$="" THEN 165
170 A=ASC(A$)-65
180 IF A<0 OR A>2 THEN 160
190 PAUSE 50:LOCATE 2,1:PRINT "イクツ トリマスカ
    ";
200 B$=INKEY$:IF B$="" THEN 200
205 B=VAL(B$)
210 IF B<1 OR N(A)<B THEN 160
220 N(A)=N(A)-B:GOSUB 1000
230 GOSUB 2000:IF EN=1 THEN P=1:GOTO 3000
300 PAUSE 90:LOCATE 2,1:PRINT"2 バン ノ ヒト ド ノ サラカラ
トリマスカ";
310 A$=INKEY$:IF A$="" THEN 310
320 A=ASC(A$)-65
330 IF A<0 OR A>2 THEN 300
340 PAUSE 50:LOCATE 2,1:PRINT "イクツ トリマスカ
    ";
350 B$=INKEY$:IF B$="" THEN 350
360 B=VAL(B$)
370 IF B<1 OR B>N(A) THEN 340
380 N(A)=N(A)-B:GOSUB 1000
390 GOSUB 2000:IF EN=1 THEN P=1:GOTO 3000
```

山の中に残っているように取れば，相手に最後の１個を取らせることができます。この数字の列を「必勝系の数列」と呼びます。相手がＫ個以下の小石を取ることができる（ただし，０個は許されない）時に，この必勝系は，（Ｋ＋１）×Ｎ＋１（Ｎは０，１，２……）となります。これは，上に書いたルールの時の必勝系で，山の数や，１度に取る山の数等が変われば，当然必勝系はもっと複雑に変化します。当然，ゲーム自体ももっとむずかしくなります。

★遊び方

このプログラムは，ゲームの中身を良く知ってもらうために，２人ゲームとなっていますが，１人ゲームのための，必勝系の見つけ方は後で書いておきますので，プレイヤーを１人けずって，そこを書き直してコンピューターにプレイさせる場合の参考にしてください。

先手は，プログラムをキーボードより，コンピューターに入れなければなりません。ここでこのプログラムにかぎらず，プログラムリストの入力は正確にやりましょう。スペース１個のちがいでもプログラムが走らなくなることもありますから…このゲームは２人用にしてありますので，コンピューターのやる仕事は２人のプレイヤーがルールにしたがっているかをチェックすることです。

Ｆ８キーを押してプログラムを動かしましょう。先ず，先手のプレイヤーに，どの山から玉を取るかを聞いてくるので，Ａ，Ｂ，ＣのどれかのKEYをたたきます。次に，取る玉の数をKEY　INするのですが，１つの山から玉を10個取ることはできません。１個から９個の間でその山にある玉の個数以内ならばいくつでも取れるはずです。KEYを押すと，コンピューターは，その山の玉をこちらが決めただけ取ってくれます。次に後手のプレイヤーの番です。KEYの使い方は先手と同じで，ＡＢＣと123…のKEYを順に押すだけでよいのです。前に書いたように最後にのこった１個を取った人のマケです。

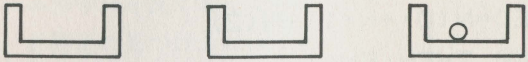
ところで，このゲームの必勝法について書いておきましょう。必勝と言うぐらいですから必ず勝てる方法があるのです。でも必ず勝てると言うことは，ゲームにならないと言うことですね。でも大丈夫です。自分の頭で考えていたら，何分も（何時間も？）かかるはずですから。このゲームのように，山が３つもある時には頭の中だけで考えることはむずかしいのです。そこで図に書いて考えましょう。

まず３つの山にのこっている玉を，それぞれＡ個，Ｂ個，Ｃ個とします。Ａの山が10個だとすると，それぞれを１，２，４，８の３つの数の和のかたちであらわします。たとえばＡ，Ｂ，Ｃが10，７，６だとするとＡは８と２の和，Ｂは４と２と１の和，６は４と２の和になります。ＡＢＣの中で，１を持っているのはＢのみ，２を持っているのはＡＢＣ，４を持っているのはＢとＣ，８を持っているのはＡだけです。ここで，１，２，４，８について，Ａ，Ｂ，Ｃ全部が同じ物を持っている時か，同じ数を持っているのがＡかＢかＣの１つだけの時

```
400 GOTO 160
1000 FOR SL=0 TO 2:FOR TM=0 TO 9
1010 LOCATE PX(SL)+DX(TM),18-DY(TM)
1020 IF TM+1<=N(SL) THEN  PRINT CHR$(&HCF);:GOTO 1
040
1030 PRINT " ";
1040 NEXT:NEXT
2000 EN=0:ED=N(0)+N(1)+N(2)
2100 IF ED=0 THEN EN=1
2200 LOCATE 2,5:PRINT "A ﾉ ｻﾗ";
2300 LOCATE 11,5:PRINT "B ﾉ ｻﾗ";
2400 LOCATE 20,5:PRINT "C ﾉ ｻﾗ";
2500 RETURN
3000 LOCATE 2,1:PRINT P+1;" ﾊﾞﾝ ﾉ ﾏｹ":PAUSE 500
3100 CLS:LOCATE10,2:PRINT "HIT ANY KEY";:PAUSE:RUN
6000 NN=((N(0) XOR N(1)) XOR N(2))
6010 NN=(NOT NN):NN=(NN AND 15)
6020 IF NN<>15 THEN NI=0:NN=((NOT NN) AND 15):GOTO
 6900
6030 NI=1
6900 RETURN
10000 DATA 00,00,01,00
10010 DATA 02,00,03,00
10020 DATA 00,01,01,01
10030 DATA 02,01,00,02
10040 DATA 01,02,00,03
10050 DATA 03,12,21
```

を“必勝形”と言います。次の玉の取り方によってこれを“必勝形”でなくすることが出来ます。つまりいま，必勝形でなければ，次に取る人が必勝形に出来る（次の人が必ず勝つということです）A，B，C，に玉が1個ずつあったとします。このとき，A，B，Cすべてが同じ数1を持っているので，これは必勝形です。同じようにA，B，Cが2個ずつ持っている時が必勝形であることを自分でたしかめてみてください。

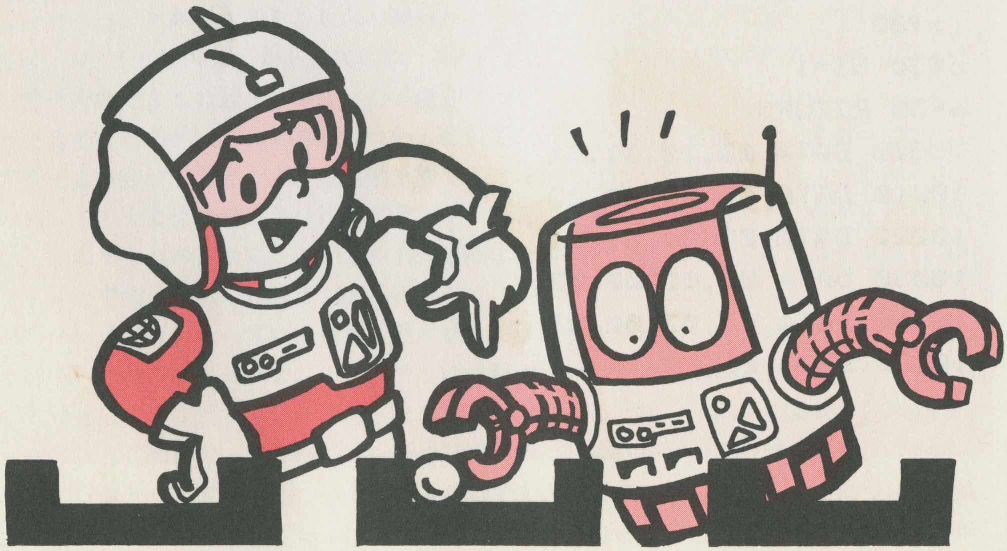
目標にする型は１つのさらに１つの玉を残すことです。



今の数をＡが６個，Ｂが５個，Ｃが７個とすると。

	Ａは	Ｂは	Ｃは
４個	○○○○	○○○○	○○○○
	＋	＋	＋
２個	○○		○○
		＋	＋
１個		○	○

ＡＢＣは４個のまとまりを共通に持っているから必勝形。



磁石こわいよ！

プログラマーの多くは磁石恐怖症だ。

これはプログラムの保存法であるカセット・テープやディスケット（黒い四角なもので中にレコードのような丸いものが入っている）共に，磁気を使って記録しているからだ。

音楽を録音したカセットテープに強力な磁石を近づけると，中の音楽が消えたり雑音になってしまうという経験はないだろうか？つまり，磁石はそういう磁気で記録したものをぶっこわしてしまうんだ。

何も磁石とは限らない。テレビや，モーター，スピーカーも強力なものだとやはりテープやディスケットの記録をこわしてしまう。

あるプログラマーがいた。彼には非常にやさしい恋人がいた。週に一度，彼女は彼の部屋にあそびにきて，ちらかった物をかたづけてくれる。

ところが，彼が仕事につかっているディスケットの中身が時々こわれてしまうのだ。注意していると，どうやら彼女があそびにくるたびに何枚かのディスケットがこわれるらしいのだ。

犯人は彼女がつけだした「健康マグネット指輪」だった。ちらかっているディスケットを箱にいれる時，指につけたマグネット指輪の磁石がディスケットの記録を破壊していたんだ。

ある会社員には３歳の女の子がいた。まだパソコンを買ってまもなく，ディスケットがこわれるのは，自分の操作ミスだと思っていた。

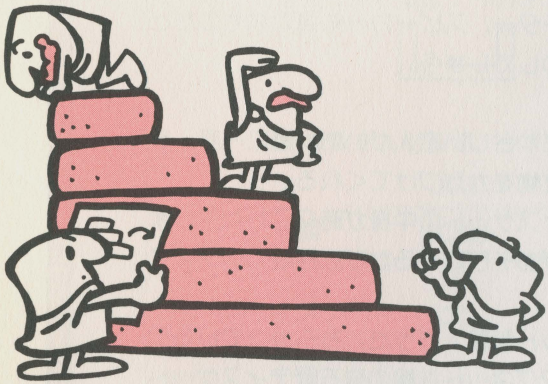
ところがある日，娘のあそんでいるのを見てびっくりした。女の子は××ちゃん人形をディスケットの上にのせ，ていねいにグルグルまわしていたのだ。もちろん，その人形の足の部分には磁石がついていた。

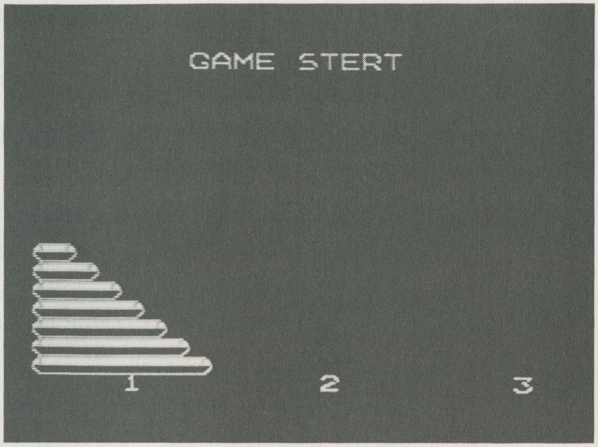
プログラマーがなぜ磁石恐怖症になってしまうかわかるね。

ディスケットにくらべテープの方が磁石の影響をうけにくいけど，注意することにこしたことはない。弱い磁力でも長時間だとおかしくなるからね。

学校で実験につかう磁石や，健康のための磁気マットや磁気マクラなんかも要注意だ！

ハノイの塔





★ゲームの説明

これは，インドに実際にあるもので，3本の柱にドーナッツ状のリングが，大きいものから順につみ上げられていて，64枚の板があります。それをすべて他の2本の柱のどちらかへ移し変えると，世界は終わるというものです。ただし，1枚ずつ移し変え，しかも小さい板の上には大きい板をおくことができません。

そのために，たとえば2枚の板について考えてみましょう。3本の柱を〔A〕，〔B〕，〔C〕としてリングの小さい方を〔1〕，大きい方を〔2〕とします。まず〔A〕の柱に下から〔2〕，〔1〕とリングがのっています。それを〔B〕の柱に移すとします。〔1〕を〔A〕から〔B〕に移してみましょう。そうすると，〔A〕にまだ残っている〔2〕は〔B〕には移せません。（小さいリングの上には大きいリングは移せないからです。）

では〔A〕の柱の〔1〕，〔2〕はどうすれば〔B〕の柱に下から〔2〕，〔1〕となるように移せるでしょうか。それは，まず〔1〕を〔C〕の柱に移し，〔2〕を〔B〕に移します。そこで，ようやく〔C〕の柱の〔1〕を再び〔B〕に移すとでき上りです。

ではリングが3枚ならどうでしょう。今度は上2枚のリング〔1〕，〔2〕を〔C〕の柱に移し，残った〔3〕を〔B〕の柱に移し，再び〔1〕，〔2〕を〔B〕の〔3〕の上に移

プログラムリスト

```
10 SPRITE OFF:CGSET 1,0:CLS
100 LOCATE 6,3:INPUT "ﾅﾝﾀﾞﾝﾆ ｼﾏｽｶ";DN:DN$=STR$(DN)
:IF DN>7 OR DN<3 THEN 100
120 DIM D$(DN),DD(DN-1,2)
130 D$(0)="          ":FOR I=1 TO DN:D$(I)=CHR$(&HC4
):FOR J=0 TO I-1:D$(I)=D$(I)+CHR$(&HC5):NEXT:D$(I)
=MID$STICK(D$(I),1,I)+CHR$(&HC6):NEXT
150 FOR I=0 TO DN-1:DD(I,0)=I+1:NEXT
200 CLS:GOSUB 2000:FOR I=1 TO 3:LOCATEI*9-6,21:PRI
NTI:NEXT
220 FOR I=0 TO 4:LOCATE 7,4:PRINT"GAME STERT":PAUS
E 25:LOCATE 7,4:PRINT"               ":PAUSE25:NEXTLO
CATE9,3:PRINT"0STEP"
230 LOCATE7,5:GOSUB 1000:K1=K-1:PRINTK1+1;"ｦ";:GOS
UB 1000:K2=K-1:PRINTK2+1;"ﾍ":I0=0
240 IF I0=DN THEN BEEP:GOTO230
250 IF DD(I0,K1)=0 THEN I0=I0+1:GOTO240
260 M1=DD(I0,K1):I1=0
270 IF I1=DN THEN 300
280 IF DD(I1,K2)=0 THEN I1=I1+1:GOTO 270
290 IF DD(I1,K2)<=M1 THEN BEEP:GOTO 230
300 DD(I0,K1)=0:DD(I1-1,K2)=M1
310 GOSUB 2000:ST=ST+1:LOCATE8,3:PRINTST;"STEPS":L
OCATE 7,5:PRINT"        ":I0=0:I1=0
320 IF I0=2 THEN 360
330 IF I1=DN THEN 360
340 IF I1+1=DD(I1,I0+1) THEN I1=I1+1:GOTO330
350 I0=I0+1:IF I0=2 THEN 230
```

せばよいでしょう。

4枚以上でも，この方法をつみ上げていけばできます。

では64枚のリングをどうして他の柱に移さないのでしょうか？世界が終るのがいやだからでしょうか？それを考える前にいったい何回リングを移せばいいのか考えてみましょう。

まず2枚のときは〔A〕の〔1〕を〔C〕へ，〔A〕の〔2〕を〔B〕へ，〔C〕の〔1〕を〔B〕へと移し，3回でおわりです。

3枚では，上の〔1〕〔2〕を〔C〕に移してから〔B〕に〔3〕を移し，再び〔1〕〔2〕を〔B〕に移すわけですから，2枚のリングの移動を2回，〔3〕のリングの移動を1度行なうわけですから，(3＋3)＋1＝7回でおわりです。4枚では同じように考えて(7＋7)＋1＝15回というふうになります。

5枚では，15×2＋1＝31回。6枚では，31×2＋1＝63回。7枚では，63×2＋1＝127回。1枚リングがふえるだけで倍の時間プラス1回ということになります。これを，N 枚について回数を数えると，2^N-1 回（2^N というのは2を N 回かけるということです。たとえば，2^3 は2×2×2のことで，2^4 なら2×2×2×2となります。）になります。

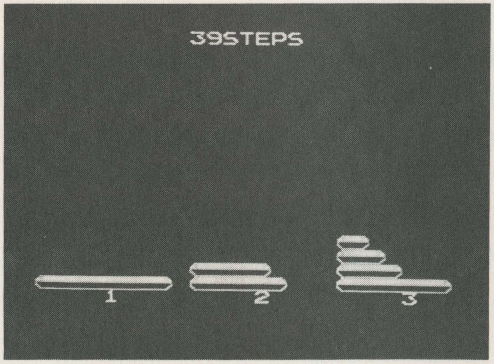
64枚だと $2^{64}-1$ 回ですから，2を64回もかけて1を引いた回数ということになり，とてもとても大きな数になります。かりに1秒に一回動かして，一度もまちがえないで，夜も寝ないでやったとしても，約六千億年もかかることになります。もう世界の終わりがどうのこうのと言っている場合ではありませんね。

そういうわけで，画面の表示のことも考えて，このゲームでは，7枚までのリングの移動ができるようにしてあります。3枚以下もすぐにできてしまうので，できないようにしてあります。移動方法は，1番2番3番の柱から，どの柱に移すかを1から3までの数字でキーボードを打ってください。たとえば，1番の柱から3番の柱へ移すときは，1のキーを打ってから3のキーを打てば，1から3へ1つだけリングが移動します。2の柱から3の柱へ1の柱にあったとおりに移し終えると，ゲームは終わりです。

もう少しリングを増したいときには，メモリーも，まだあまっているので，表示にくふうをすれば，もう少しなら，可能です。

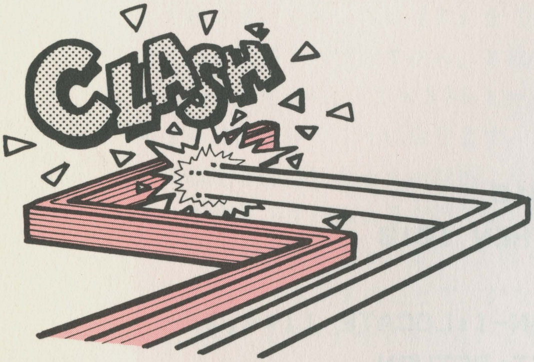
ゲームが終わると点数が出るのですが，それは，前に書いた 2^n-1 回が一度もまちがわずにできる回数ですから，かかったステップ数でそれを割った値を100分率で出しています。したがって100点満点になっています。

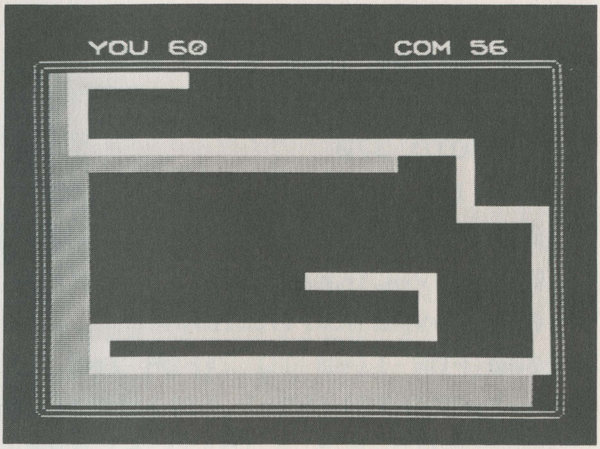
```
355 I1=0:GOTO 330
360 BEEP:BEEP:BEEP:FOR I=1 TO DN:READ S2:NEXT:LOCA
TE4,8:PRINT"GAME END":LOCATE 14,8:PRINT"トクテン";S2*1
00/ST;"テン":
370 LOCATE3,10:PRINT"モウイチド ヤリマスカ 1-ハイ 2-イイエ":GOSU
B 1000
380 IF K=1 THEN RUN
390 IF K=2 THEN END
400 GOTO 370
1000 IF INKEY$<>"" THEN 1000
1010 K$=INKEY$:IF K$="" THEN 1010
1020 K=VAL(K$):IF K<1 OR K>DN THEN 1010
1030 K=VAL(K$):RETURN
2000 FOR II=0 TO 2:FOR I=0 TO DN-1:LOCATE II*9,21-
DN+I:PRINTD$(DD(I,II)):NEXT:NEXT:RETURN
3000 DATA 1,3,7,15,31,63,127
```





CLASH LINE





このゲームは，四角で囲まれた画面の中で，勝手にのびていく棒（LINE）を，ぶつからないように，ジョイスティックを使って方向を変えるゲームです。

自分は白の棒，コンピューター側は青の棒がのびて行きます。それを，ジョイスティックで方向を変えて，わく線（外側の二重線）または自分の線やコンピューターの線につき当たらないようにします。また，進行方向と逆方向（右に進んでいる時左方向，上に進んでいる時下方向）に進もうとすると，ゲームオーバーです。コントローラーはIのみ使用です。トリガーボタン［A］を押すと，第2回戦スタートです。ゲームは“YOU”何勝，“COM”（コンピューター）何勝というふうに，画面上に表示されます。（はじめは，何回勝負で，“YOU”の勝ち，“COM”の勝ちとしたり，最初に何勝した方の勝ちとしたのですが，それでは，どうも，まどろっこしいので，やめました。すごく，熱中するのです。コンピューターなんかに，負けてたまるか，などと思いつつ，けっきょく勝ち数，何対何で，いくつ勝っているな，とか負けているな，とか表示することにしました。）

★プログラムの説明

プログラムの内容は，15行で，“YOU”，”COM”を表示し，20行から40行で外側のわく線を書きます。この部分の書くスピードが，おそいと思う場合は，BG グラフィックを，打

プログラムリスト

```
10 SPRITE OFF:CGSET 1,0
15 CLS:LOCATE 3,0:PRINT "YOU";YT:LOCATE 19,0:PRINT "COM";CT
20 FOR I=1 TO 26:LOCATE I,1:PRINT CHR$(227);:LOCATE I,22:PRINT CHR$(227);:NEXT
30 FOR I=2 TO 21:LOCATE 0,I:PRINT CHR$(226);:LOCATE 27,I:PRINT CHR$(226);:NEXT
40 LOCATE 0,1:PRINT CHR$(222);:LOCATE 27,1:PRINT CHR$(223);:LOCATE 0,22:PRINT CHR$(224);:LOCATE 27,22:PRINT CHR$(225);
50 S=RND(4):S1=1:FOR I=0 TO S:S1=S1*2:NEXT:S1=S1/2:S=RND(4):S2=1:FOR I=0 TO S:S2=S2*2:NEXT:S2=S2/2
60 X1=RND(15)+7:Y1=RND(10)+7:X2=RND(15)+7:Y2=RND(10)+7:IF ABS(X1-X2)<4 AND ABS(Y1-Y2)<4 THEN 60
100 ST=STICK(0):IF ST<>0 THEN S1=ST
110 GOSUB 800:IF CH=0 THEN DE=0:GOTO 2000
120 X1=X1+X:Y1=Y1+Y:LOCATE X1,Y1:PRINT CHR$(253);
200 GOSUB 910:IF CH=0 THEN TA=1:GOTO 250
210 GOSUB 900:IF CH=0 THEN TA=2:GOTO 250
220 X2=X2+X:Y2=Y2+Y:LOCATE X2,Y2:PRINT CHR$(254);:GOTO 100
250 IF S2>3 THEN 400
300 F=0:S=4:SP=1:GOSUB 940:IF CH=0 THEN F=1
310 S=8:SP=1:GOSUB 940:IF CH=0 THEN F=F+2
320 IF F=0 THEN S2=(RND(2)+1)*4:GOSUB 910:GOTO 220
330 IF F=1 THEN S2=8:GOSUB 910:GOTO 220
340 IF F=2 THEN S2=4:GOSUB 910:GOTO 220
350 IF TA=1 THEN DE=1:GOTO 2000
```

ちこんで，20行をとり，15行を，

15 CLS：LOCATE ……… から，

15 CLS：VIEW：**LOCATE** ………

というふうに，CLSとLOCATEの間に，VIEWを入れ，30行を，

30 FOR I＝**2** TO ……… から，

30 FOR I＝**21** TO ………

というふうに，I＝2を，I＝21に変えてください。少しは速くなります。

50行，60行は，線の出てくる座標，進んで行く方向を決めています。変数S1，S2は，自分とコンピューターの進む方向で，ジョイスティックのSTICK関数の値と同じ値をもちいています。方向と数値の関係を知りたければ，取扱説明書のSTICKの項目を見てください。60行のIFは，自分とコンピューターのスタートの位置が近い場合，やり直すようにしています。

100行から120行は，自分が，ジョイスティックの入力を行なったときに，線の進む方向を変え，これから進む方向に何もしないことを確認して，画面にプリントします。何か障害物があるようだと，ゲームおわりのルーチンに行きます。（2000行から2040行）

200行から460行までが，コンピューターの方向をきめるルーチンです。

まず，進む方向の直前に，障害物があるかどうかをしらべて，あるときは，その両側をしらべ，行ける方向へ曲ります。（両側とも行けるときには，乱数できめます。）行けないときには，ゲームおわりのルーチンに行きます。進行方向直前に何もないときには，もう一つ手前をしらべて，同じように曲ります。（ただし，両側とも行けないときには直進します。）**300行から360行**までが縦方向をしらべるルーチン，**400行から460行**までが，横方向をしらべるルーチンです。このプログラムでは，似たようなルーチンを300行からと400行からに分けて書きましたが，一つにまとめたり，または，サブルーチンでうまく処理したりして，短くて速いルーチンにすることも可能でしょう。**800行から1060行**までは，ジョイスティックのSTICK関数に対応する方向をしらべ，その場所に，何か障害物があるかないかをしらべるためのサブルーチンです。このサブルーチンは，自分のSTICK入力の判断と，コンピューターの進行方向の判断を，同時にできるように，自分のときは800行から入り，コンピューターのときは，900行，910行，940行から入るように作ってあります。また，コンピューターのときに，3ヵ所の入り口をもうけたのは，それぞれの入り口で，まず900行のときは，2つ手前をしらべるようになっていて，910行のときは，一つ手前で，940行では，しらべる方向の変数Sの値をあらかじめ決めて（310行や410行など），すきな方向についてしらべられるるようにしてあります。このようにすると，ふつうのサブルーチンだと一種類の処理しかできませんが，このように一たん別の所（800行から940行）へとばしてからサブルーチン本体（1000行から1060

```
360 GOSUB 910:GOTO 220
400 F=0:S=1:SP=1:GOSUB 940:IF CH=0 THEN F=1
410 S=2:SP=1:GOSUB 940:IF CH=0 THEN F=F+2
420 IF F=0 THEN S2=RND(2)+1:GOSUB 910:GOTO 220
430 IF F=1 THEN S2=2:GOSUB 910:GOTO 220
440 IF F=2 THEN S2=1:GOSUB 910:GOTO 220
450 IF TA=1 THEN DE=1:GOTO 2000
460 GOSUB 910:GOTO 220
800 XX=X1:YY=Y1:S=S1:SP=1:GOTO 1000
900 SS=1
910 SP=SS+1
930 S=S2
940 XX=X2:YY=Y2:GOTO 1000
1000 IF S=1 THEN X= 1:Y= 0:GOTO 1040
1010 IF S=2 THEN X=-1:Y= 0:GOTO 1040
1020 IF S=4 THEN X= 0:Y= 1:GOTO 1040
1030 IF S=8 THEN X= 0:Y=-1:GOTO 1040
1040 IF SCR$(XX+X*SP,YY+Y*SP)=" " THEN CH=1:GOTO 1
060
1050 CH=0
1060 SS=0:RETURN
2000 BEEP:IF DE=1 THEN YT=YT+1:LOCATE 6,0:PRINT YT
;:GOTO 2020
2010 CT=CT+1:LOCATE 22,0:PRINT CT;
2020 IF STRIG(0)<>8 THEN 2020
2040 GOTO 15
```

行）へ入るようにすれば，入り口を変えるだけで，あるていど異った処理ができるサブルーチンになります。また1000行から1030行などは，変数をつかってX，Yの値を代入しておいて，Sの値を添字に用いて行なうと，ある程度わかりにくいプログラムにはなりますが，コンパクトなプログラムとなります。

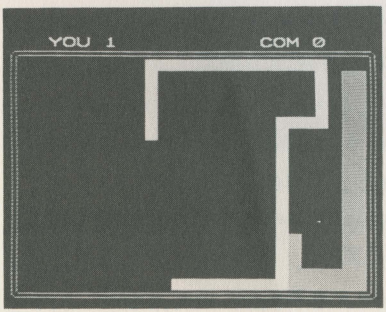
2000行からは，自分とコンピューターの勝ち数を表示するルーチンです。

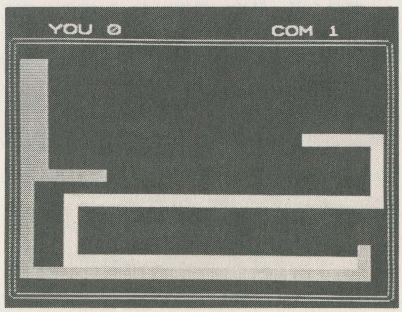
★プログラムの改造

これまでに，プログラムの簡単な説明をしましたが，それでわかるように，コンピューター側の動きは，基本的には障害物がなければどこまでも直進します。そして2つ手前に何かあれば曲がるようになります。なぜ1つ手前だけでなく2つ手前でも曲がるようにしたのかというと，このゲームは，やってみるとわかると思いますが，LINEとLINEのすきまにうまく相手を入れて，ふくろ小路にとじこめてやっつける，というテクニックが，一番の必勝法なのです。そのために，まちがえることのないコンピューターにはこの必勝法だけで十分に強くなっているはずですし，これがないと，動きが単調になり，しかも弱くなります。この他にも必勝法はいくつかありますが，あまり多くのテクニックをコンピューターに，覚えこませる（プログラミングしておく）と，それを考える分だけおそくなり，こちら側が考える時間がふえますし，方向がえのタイミングもまちがえることが少なくなります。それではあまり負けなくなってしまってつまらないものになるでしょう。それから，ファミリーベーシックのフリーエリアとの関係もあり複雑なものはむずかしいでしょう。

また，コンピューター側のルーチンをとって，もう一つコントローラー2の方のルーチンをつけ加えれば，2人用のゲームとしても楽しめるでしょう。少しコンピューターの動きが速いとか思われる場合は，100行のST＝STICK（0）の前にPAUSE文を入れ変数を適当に決めておけば，自分に合ったスピードで，行なえるでしょう。

最後に，いろいろとテクニックがあるようですから，ためしてみてください。それにあきたら，改良してもっと速くするなり，楽しんでみてください。





職業としてのプログラマーをめざす人たちへ

おやめなさい！

と言ってしまうと話しにもならないね。

プログラム自体は最高だ。何もない所から新しいなにかを築きあげてゆく。完成した時の喜びは言葉ではいいあらわせない喜びがある。

でも，大人になって，職業にしてしまうのはどうかな。

たしかに給料は良いみたいだし，何より，日本はプログラマーの不足で，大学や専門学校を出れば就職にはこまらない。

どんな職業もみんな同じかもしれないけど，いざ仕事としてプログラマーというと，けっして良いもんじゃないよ。

まず，残業はあたりまえ。デートの約束なんてそっちのけ。特にしめきり近くになると徹夜の連続。当然，おふろへ行く時間ももったいなくなる。

食事は不規則になるから，やせてくるか，逆に仕事中ポリポリとおかしを食べて太ってくるかのどっちかだ。

メガネはプログラマーのトレードマーク。

頭の中はプログラムの事で一杯。道路を歩いていてもうわの空だから，下手すりゃ交通事故。

神経ズタズタ，体はボロボロ，恋人や奥さんには逃げられ（冗談じゃなく，アメリカではマイコン・ウイドウ）なんて新しい言葉まであるんだ。主人がコンピューターにかかりっきりで，まるで未亡人みたいな奥さんという意味だ）家族はソッポむく。

それでもプログラミングの喜びがあればいいけど，大きな会社になると，「企業秘密」の名のもとに，自分が何を作っているかも知らされずに，上司の言うままのプログラムを作らされる。

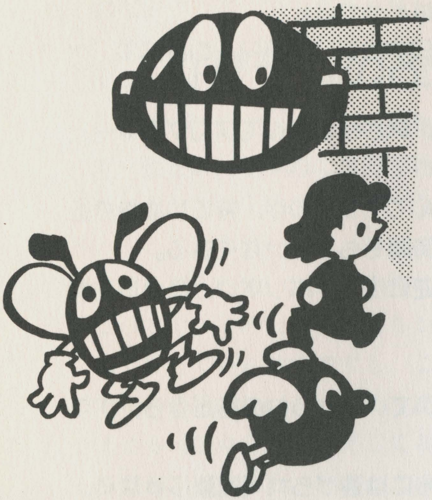
ゲームを作りたいのに何がなんだかわからないものを作らされる。できあがったプログラムは会社のもので，作った本人にはなんの権利もない。

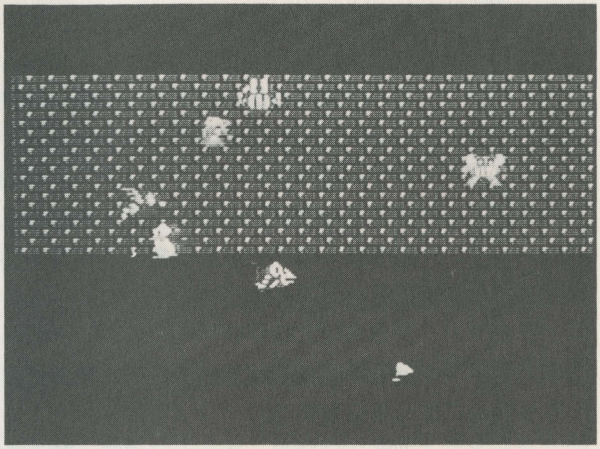
やっぱり，マイコンやプログラムは趣味のままがいいね。仕事じゃなきゃ，つかれたらあそびに行けばいいし，しめきり，なんて恐怖もなく，マイペースでのいじりプログラミングを楽しめるしね。

だから，もう少し，プログラマーの生活環境がよくなるまでは，職業としてのプログラマーは考えなおした方がいいよ！

えっ？　私？　えー，やっぱり，プログラマーを続けますよ！

パーソン・スナッチ





日も，とっぷりと暮れて，星も無い晩ともなると，街を通る人の中には，物に取りつかれた目を見ることがある。夜遅く，近くにそんな急いで行くような所も見当らないのに，何かを思いつめたような人が，顔に氷ついたような冷たい笑いを浮かべて道を急ぐ。足は動いていないのに，すべるように進む。寒々とした林の上に現われた，首切り鎌のように細い三日月の光が，その顔を照らす。もはや人間とは思えぬ顔がちらと見え，また，闇の中に消える。冷たい水をかぶったような寒気。青白い顔，見開かれた焦点の合わぬ目。それは何もない虚の空中をにらんだまま，血の赤を程している。スナッチャーに乗っ取られた人間の姿である。

奴が何を求めて，どこへ急ぐのかは知らない。ただ，解っているのは，奴が実際に存在することと，身の回りの誰がそうなっても不思議は無いということだけだ。

日も，とっぷり暮れて，星も無い晩に道を歩いていたら，気を付けた方が良い。いきなり気が付くと，わけの解らない所にいることがある。これは，君がしばらくの間，スナッチャーに乗っ取られていたためだ。君は，意識を失なう直前に，すれ違った人に宿っていたスナッチャーに乗り移られたのだ。気がついた時に，奴はまた，誰かに乗り移って，どこかへ行ってしまったのだ。

街中で，君は知らぬ間に，氷ついたような笑いをその顔に浮かべ，歩いているかもしれない。今は昼？夜？今夜は三日月かい？君は夜，外に出かけなければならぬ用がある？

プログラムリスト

```
10 CLS:VIEW
100 DEF MOVE(0)=SPRITE(11,3,1,255,0)
110 DEF MOVE(1)=SPRITE(11,3,1,1 ,0)
120 DEF MOVE(2)=SPRITE(11,7,1,1 ,0)
130 DEF MOVE(3)=SPRITE( 1,3,1,255,0,2):POSITION3,0
,72
140 DEF MOVE(4)=SPRITE( 2,7,2,255,0,2):POSITION4,1
00,88
150 DEF MOVE(5)=SPRITE( 3,3,3,255,0,3):POSITION5,2
00,104
160 DEF MOVE(6)=SPRITE( 4,7,4,255,0,2):POSITION6,1
50,120
170 DEF MOVE(7)=SPRITE(15,3,4,255,0,2):POSITION7,2
50,136
180 DEF SPRITE 0,(1,1,0,0,0)=CHR$(88)+CHR$(89)+CHR
$(90)+CHR$(91)
190 MOVE 3,4,5,6,7:SPRITE ON
195 ERA 0,1,2
200 X=100:Y=56
205 ST=STICK(0)
210 GOSUB 1000:IF (ST AND 3)=0 THEN 240
220 IF ST=2 THEN X=X-1
230 IF ST=1 THEN X=X+1
240 SPRITE 0,X,Y
300 ST=STRIG(0):IF ST=0 THEN 205
310 POSITION 1,X,Y:P1=1:P2=2:NO=2:MOVE1:POSITION2,
X,Y:MOVE2
400 SPRITE ON
```

★遊び方

スナッチャーの正体は，通称ニタニタです。彼は，画面の下の旗へ行きたがっているのです。そのために，人を乗っ取って，乗り移って，その旗をめがけて，青い顔に，〝ニタニタ笑い〞を浮かべて進んでゆきます。

プログラムと，BG グラフィックのデータを打ち込んで RUN すると，画面上方に，ニタニタが現れます。ここで，コントローラーIの右，及び左のジョイスティックを押すと，ニタニタは，それぞれ右，左に動きます。位置を見定めて，トリガーAを押すと，ニタニタはキャラクターに乗り移り出します。

画面中央の動いているキャラクターは，街を歩いている人々です。ニタニタは，下へ行きたい一心で，最初にすれ違うキャラクターに乗り移り，次のキャラクターとすれ違うと，また同じように乗り移ってゆきます。こうして，5つのキャラクターを次々に乗り移った後，ニタニタは，真下に向けて突進します。最初のニタニタの位置と，トリガーAを押すタイミングを調整して，ニタニタをうまく旗の所まで導いてやってください。

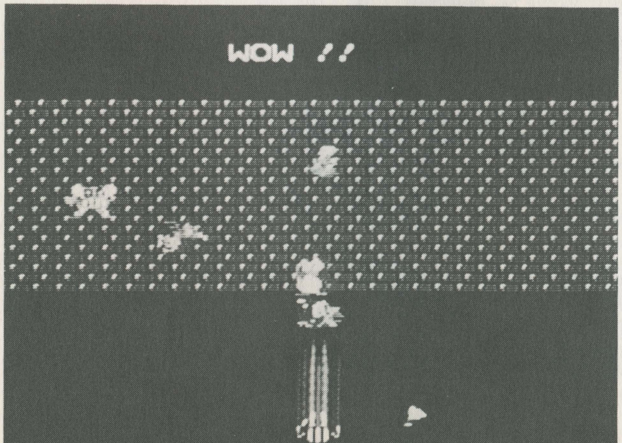
旗まで行きつくと，〝GOOD!!〞と表示が出，旗が無いと〝WOW!!〞とニタニタは驚いてしまいます。

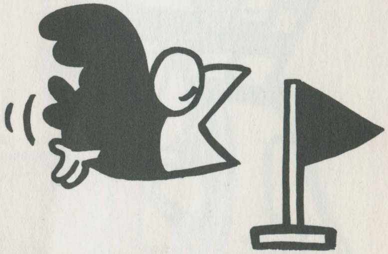
動いているキャラクターをじっと見つめて，どうすれば旗まで行きつけるか考えてください。

★プログラムの説明

ゲームの考え方は，穴の開いたベルトコンベアーが5本あるようなものと，思ってください。プログラムの中心は，ニタニタが下のキャラクターに乗り移る所にあります。実は，この部分は1行しかありません。どの部分だと思いますか？　ファミリーコンピューターでプログラムを一度でも作ったことのある人ならば，MOVE コマンドをどういった場面で使うか悩んだだろうと思います。このプログラムは，MOVE コマンドを活用することから出来上がったものなのです。MOVE を使って他の BASIC プログラムでは書けないようなプログラムを書けないかな？と考えたと思います。まずたいせつなことは，MOVE は1度し動か始めると，プログラムと同時に，ひとりでに動いてくれることです。これは，“割り込み”と呼ばれる技術を使っているためですが，見かけ上 MOVE を動かすコンピューターと BASIC プログラムを動かすコンピューターが2台あるのと同じことから，とても便利なものです。しかし MOVE はキャラクターを動かす方向と長さが，一度定めた値を変えられないために，いざプログラムを組もうとすると，使いにくいのです。そこで，ここでは“動きを変えなくても良いプログラムにしよう!!”と始めに決めてからゲームを作りだしたのです。プログラム

```
500 IF NO=7 THEN DEF MOVE(0)=SPRITE(11,5,1,24,0):E
RA P1,P2:POSITION0,XPOS(P1),YPOS(P1):MOVE 0:GOTO 2
000
510 IF XPOS(NO+1)>(XPOS(P1)-4) AND XPOS(NO+1)<(XPO
S(P1)+4) THEN SPRITE0:GOSUB8000:ERA P1:POSITION P2
,XPOS(NO+1),YPOS(NO+1):MOVE P2:SWAP P1,P2:NO=NO+1
550 IF (NO MOD 2)=1 THEN DR=7:GOTO 590
560 DR=3
590 GOSUB 1000:GOTO 500
1000 FOR A=3 TO 7
1010    IF MOVE(A)=0 THEN MOVE A
1020 NEXT:RETURN
2000 IF XPOS(0)>120 AND XPOS(0)<136 THEN A$="GOOD
!!":GOTO 2200
2100 A$="WOW !!"
2200 LOCATE 10,1:PRINT A$:PAUSE 500:LOCATE 10,1:PR
INT "           ":RUN
8000 DEF MOVE(P2)=SPRITE(11,DR,NO-1,255,0):RETURN
```

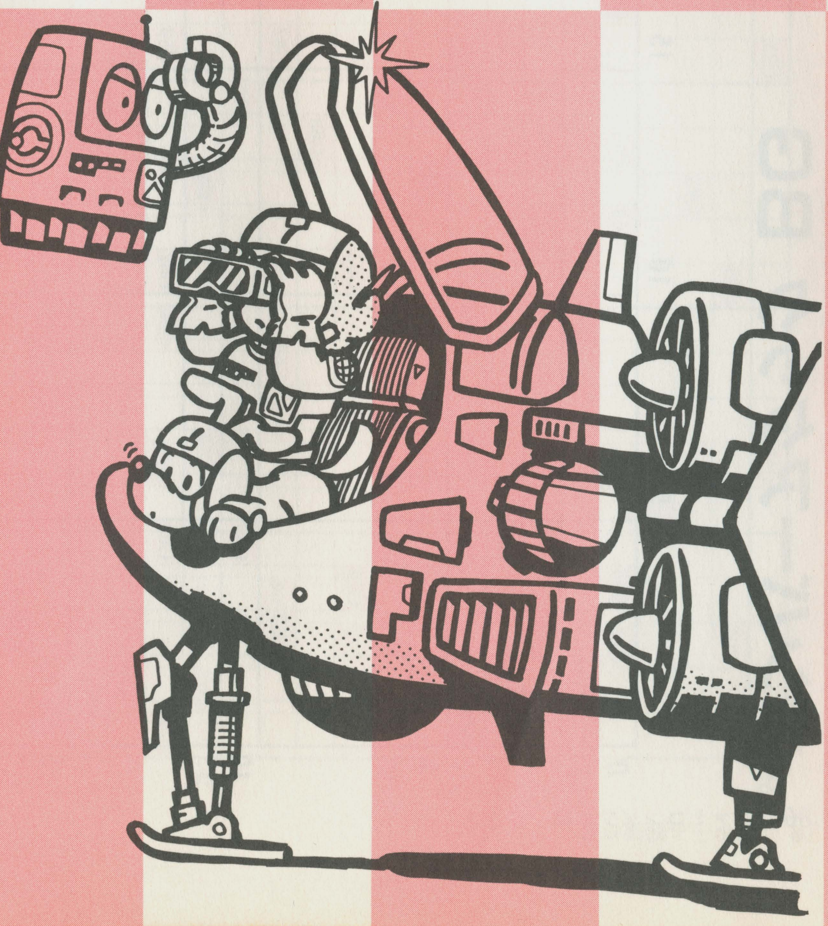




の大きさが小さくなったのもこのMOVEのおかげです。横方向に動くキャラクターはすべてMOVEコマンドを使っていますから，プログラムは上下のキャラクターが重なったかどうかをいつも見ているだけでよいのです。

MOVEで動いているキャラクターの位置は，もちろん，XPOS（NO），YPOS（NO）で知ることが出来ますから，上と下のXPOS（　）が同じになった時に，ニタニタを下に動かすプログラムを作ります。IF文で，XPOS（NO）とXPOS（NO＋1）（NOは上から何段めのキャラクターかを示しています）が同じになった時ニタニタを一度消し（ERAで消す）もう一度DEF　MOVEをやり直して下の段の位置，XPOS（NO＋1），YPOS（NO＋1）に位置を合わせてMOVEで動かし直すのです。この動きは一行で書けました。プログラムが小さいということは，速く動かすためには重要なことですから，みなさんも今まで3行使って書いていたプログラムを，2行ですますにはどうしたらよいかを考えてみると，プログラムのうでが上がると思いますよ。

BG画面データ集



ボンバーアタック BG

	Ø					5					1Ø					15					2Ø					25		27
Ø				G6Ø								G6Ø					G6Ø											
		G5Ø								G6Ø														G6Ø		G5Ø		
												G52								G6Ø								
					G6Ø											G6Ø									G52		G6Ø	
	G6Ø						G5Ø			G6Ø													G6Ø					
5		G62											G5Ø								G5Ø							G62
															G6Ø													
				G6Ø				G62			G52							G52				G62			G6Ø			
														G5Ø					G6Ø									
			G52						G6Ø							G62								G52				
1Ø					G6Ø													G5Ø									G6Ø	
							G5Ø						G52								G6Ø							
					G62						G62					G6Ø									G5Ø			
		G6Ø																				G62						
														G5Ø						G52							G62	
15	G62							G62		G52														G6Ø				
			G5Ø											G6Ø				G52										
						G52															G5Ø				G52			
													G62														G62	
			G52								G52								G62			G52						
2Ø							G62									G52												

レーザーコマンド BG

	Ø					5					1Ø					15					2Ø					25		27
Ø				G6Ø								G6Ø					G6Ø											
		G5Ø								G6Ø														G6Ø		G5Ø		
												G52								G6Ø								
					G6Ø											G6Ø									G52		G6Ø	
	G6Ø						G5Ø			G6Ø													G6Ø					
5		G62											G5Ø								G5Ø							G62
															G6Ø													
				G6Ø				G62			G52							G52				G62			G6Ø			
														G5Ø					G6Ø									
			G52						G6Ø							G62								G52				
1Ø					G6Ø													G5Ø									G6Ø	
							G5Ø						G52								G6Ø							
					G62						G62					G6Ø										G5Ø		
		G6Ø																				G62						
														G5Ø						G52							G62	
15	G62									G52														G6Ø				
			G5Ø											G6Ø				G52										
						G52															G5Ø				G52			
													G62														G62	
			G52								G52								G62			G52						
2Ø							G62									G52												

ペンギンさんとカラスさん BG

	0					5					10					15					20					25		27
0																												
			IØØ	I1Ø	I1Ø	I2Ø																						
		IØØ	I1Ø	I1Ø	I1Ø	I1Ø	I2Ø																					
			IØØ	I1Ø	I1Ø	I1Ø	I2Ø																					
5																												
														IØØ	I1Ø	I1Ø	I2Ø											
10												IØØ	I1Ø	I1Ø	I1Ø	I1Ø	I1Ø	I1Ø	I2Ø									
													IØØ	I1Ø	I1Ø	I1Ø	I1Ø	I1Ø	I1Ø	I1Ø	I1Ø	I2Ø						
															IØØ	I1Ø	I1Ø	I1Ø	I1Ø	I2Ø								
15																												
20																												

逃げるなカメ BG

	Ø					5					1Ø					15					2Ø					25		27
Ø																												
	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
5	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
1Ø	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
15	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
	M6Ø	M6Ø	M6Ø																									
2Ø	M6Ø	M6Ø	M6Ø																									

ボーリング BG

	0					5					10					15					20					25		27
0	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
5	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
10	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
15	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														
20	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41	L41														

ペンギンたたき BG

	0					5					10					15					20					25		27
0																												
5		I60	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	I70		
		J20	"F"	"1"	J20	"F"	"2"	J20	"F"	"3"	J20	"F"	"4"	J20	"F"	"5"	J20	"F"	"6"	J20	"F"	"7"	J20	"F"	"8"	J20		
		J20	J30	J30	J20	J30	J30	J20	J30	J30	J20	J30	J30	J20	J30	J30	J20	J30	J30	J20	J30	J30	J20	J30	J30	J20		
		J20			J20			J20			J20			J20			J20			J20			J20			J20		
		J20			J20			J20			J20			J20			J20			J20			J20			J20		
10		J20			J20			J20			J20			J20			J20			J20			J20			J20		
		J20			J20			J20			J20			J20			J20			J20			J20			J20		
		J20			J20			J20			J20			J20			J20			J20			J20			J20		
		J00	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J10		
15																												
20																												

ルーレット BG

	Ø					5					1Ø					15					2Ø					25		27
Ø																												
		K71	K51	K51	K21	K51	K51	K21	K51	K51	K21	K51	K51	LØ1					K71	K51	K51	K21	K51	K51	K21	K51	K51	LØ1
		K61			K61			K61			K61			K61					K61			K61			K61			K61
		K61		1	K61		2	K61		3	K61		4	K61					K61		1	K61		2	K61		3	K61
		K41	K51	K51	KØ1	K51	K51	K11	K51	K51	KØ1	K51	K51	K31					K41	K51	K51	KØ1	K51	K51	KØ1	K51	K51	K31
5		K61			K61	フ	ァ	ミ	コ	ン	K61			K61					K61			K61			K61			K61
		K61	1	2	K61						K61		5	K61					K61		4	K61		5	K61		6	K61
		K41	K51	K51	K31						K41	K51	K51	K31					K41	K51	K51	KØ1	K51	K51	KØ1	K51	K51	K31
		K61			K61	ル	ー	レ	ッ	ト	K61			K61					K61			K61			K61			K61
		K61	1	1	K61						K61		6	K61					K61		7	K61		8	K61		9	K61
1Ø		K41	K51	K51	KØ1	K51	K51	K21	K51	K51	KØ1	K51	K51	K31					K41	K51	K51	KØ1	K51	K51	KØ1	K51	K51	K31
		K61			K61			K61			K61			K61					K61			K61			K61			K61
		K61	1	Ø	K61		9	K61		8	K61		7	K61					K61	1	Ø	K61	1	1	K61	1	2	K61
		L11	K51	K51	K11	K51	K51	K11	K51	K51	K11	K51	K51	L21					K41	K51	K51	K11	K51	K21	K11	K51	K51	K31
																			K61					K61				K61
15																			K61	E	V	E	N	K61	O	D	D	K61
		シ	ハ	ラ	イ			$											L11	K51	K51	K51	K51	K11	K51	K51	K51	L21
		シ	ョ	ジ	キ	ン		$																				
2Ø																												

ドッヂエム BG

	Ø					5					1Ø					15					2Ø					25		27
Ø		L3Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L7Ø						
		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø						
		L4Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	L4Ø						
		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		"B"	"L"	"U"	"E"	
		L4Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	L4Ø						
5		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø						
		L4Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	L4Ø						
		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		"R"	"E"	"D"		
		L4Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	L4Ø						
		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø						
1Ø		L4Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	L4Ø						
		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø						
		L4Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	L4Ø						
		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø						
		L4Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	L4Ø						
15		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø						
		L4Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	L4Ø						
		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø						
		L4Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	M6Ø	L6Ø	L4Ø						
		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø		L4Ø						
2Ø		L5Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	L6Ø	MØØ						

ニム BG

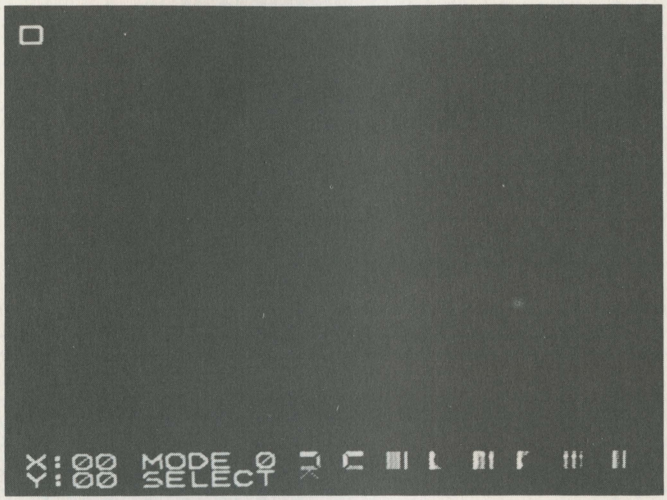
	Ø					5					1Ø					15					2Ø					25		27
Ø																												
5																												
1Ø																												
15																												
			L41					L41				L41					L41				L41					L41		
			L51	L61	L61	L61	L61	MØ1				L51	L61	L61	L61	L61	MØ1				L51	L61	L61	L61	L61	MØ1		
2Ø																												

パーソンスナッチ BG

	0					5					10					15					20					25		27
0																												
	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
5	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
10	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
15																												
20																			F72									

BG画面入力法

●写真1



ＢＧ画面データは，ＢＧ画面モードを選択してから入力します。Ver. 2ベーシックカセットなら，ダイレクトに

SYSTEM

とすると，ベーシックとＢＧ画面の選択画面になるので，数字キーの「２」を押してください（リターンキーは押す必要がありません)。この時，すでにベーシック画面でプログラムを入力していても保存されています。

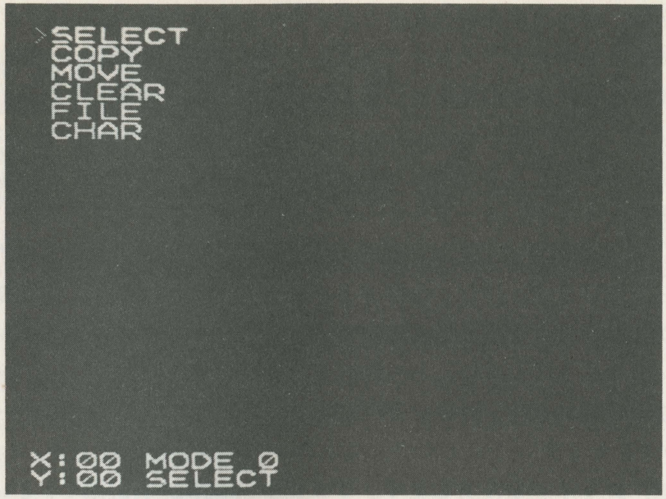
Ver. 3ベーシックカセットでは「SYSTEM」命令は他の意味を持っています。かわりにダイレクトで，

BGTOOL

と入力してください。すぐにＢＧ画面になります。

このＢＧ画面でのキャラクタ入力法には，「COPY」モードや「MOVE」モードなどの便利な機能もありますが，「ファミリーベーシック・ホップ編」にくわしく説明してありますので，そちらを参照してください。

●写真２



基本的な入力法，「SELECT」モードを説明します。ＢＧ画面の一番最初は**写真１**のような状態になっています。左上の白い四角が「カーソル」で，キャラクタを書き込む場所を意味します。この「カーソル」の位置は，画面左下にＸ座標，Ｙ座標で表示されます。

「MODE」と書かれた右に数字が出ています。これは，キャラクタの色の番号です。RETURNキーを一回押すたびに，Ø→１→２→３→Ø→１，のように変化します。次に画面右下に８つの図形が表示されています。この図形がキャラクタで，この８個をキャラクタセットといい，キャラクタテーブルＢの横一列にあたります。開始画面では，キャラクタテーブルのＡ列が表示されているはずです。「CLR・HOME」キーを押すたびに，Ａ→Ｂ→Ｃ…Ｍと，下の列が表示されます。「SHIFT」キーと，「CLR・HOME」キーを一緒に押すたびに，Ｍ→Ｌ→Ｋ…→Ａと，逆に上の列へ表示が移っていきます。

この八個のキャラクタの下（最初は一番左のキャラクタの下）に「へ」マークが出ています。これをセレクトカーソルといい，「DEL」キーを押すと右へ，「INS」キーで左へ移動します。

カーソルキーでカーソルを書き込む予定の場所へ移動して，スペースキーを押すと，その時セレクトカーソルの上にあるキャラクタがカーソルのあった場所に書き込まれます。間違えたキャラクタを消すためには，カーソルをそのキャラクタの上に持っていき，文字キーの「Ｄ」を押します。

ＢＧ画面データは「Ｅ５Ø」のように，３文字か，「“Ｔ”」のようにダブルクォーテー

ションで囲まれた文字になっています。

「E５0̸」とは，キャラクタ表（キャラクタテーブルB）の「E」列のナンバー「５」(左から６番目）にあたるキャラクタをカラーナンバー「0̸」で書く，という意味です。「“T”」のような文字を書き込むには次の操作が必要です。まず，キーボード左上の「ESC」キーを押します。画面左上に文字が現われ，一番上の文字の左に「>」マークが表示されます(**写真２**)

この「>」を「CHAR」という所に合わせてスペースキーを押すと，左上の文字が消えて，書き込みモードになります。カーソルキーでカーソルを目的の場所へ移動し，目的の文字キー（たとえば「T」)を押すと，そこに目的の文字が書き込まれます。もちろん文字を消すにはスペースキーを使います。(「D」を押すと「D」と表示されます。) **写真２**の左上の文字の下にすでに何かキャラクタを書き込んでいても，そのキャラクタは消えていません。

すべてのデータを書き終わったら，ＢＧ画面をテープにSAVEしてください。ESCキーを押して，**写真２**の状態で「>」マークを「FILE」に合わせてスペースキーを押します。

LOADかSAVEかを聞いてくるので，文字キーの「S」を押します。ファイルネームを聞いてきますので，ファイルを入力します。プログラム本体が「TEST」という名前なら，

TESTBG

などとするとわかりやすいでしょう。

SAVEが終わると，自動的に書き込みモードに変わります。もう一度ESCキーを押して，**写真２**の状態でSTOPキーを押すと，ベーシック（「１」を押す）にしたあとプログラムを打ち込んでください。一つのゲームのプログラムとＢＧ画面は続けてSAVEした方が便利でしょう。Ver. 3では，ベーシック上で直接ＢＧ画面データをLOAD, SAVEできるようになりました。

LOAD “ファイル名”

SAVE “ファイル名”

とします。また，プログラムがあまり長くない場合は，

BGGET

という命令で，ＢＧ画面のデータがプログラムのうしろに転送されるので，

SAVE “ファイル名”

とすると，プログラムとＢＧ画面データを一度にテープにSAVEできます。この方法でSAVEしたテープは，

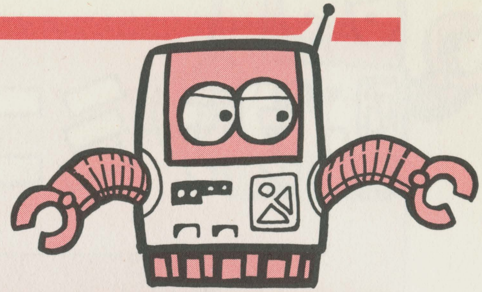
LOAD “ファイル名”

としたあと，

BGPUT

とすることで，プログラムのうしろにあるＢＧデータをＢＧ画面に転送してくれます。

プログラムが動かない時は？



プログラムをちゃんと打ち込んだはずなのに動作がおかしい，エラー・メッセージが出て止まってしまうことがあります。これは，プログラムの間違いでない限り入力ミスです。

SN，IL，TMの各エラーは入力のタイプミスです。その行に打ち間違いがないか確かめてください（ILエラーの一部も）

NF，RG，OD，UL，DD，MO，UF（Ver.3）のエラーは，入力しなければいけない命令や行やオペランド（LOCATE　Ø，Øのような命令の「Ø，Ø」に当たる部分）を打ち込んでいないか，逆に同じ命令をだぶって2回入力している時に起こります。

IL，OV，OM，SO，DZエラーは，どこかで変数の値が予定外になっていることが多く，「?OV　ERROR　IN　25Ø」と出ても，その原因はその行以前にある場合が多いのです。たとえば，

```
10  A=100
20  B=A*A
```

というプログラムの10行を「A=1ØØØ」と打ち間違うと，エラーは20行で発生します。

エラーは出ないけど，おかしな動きをする場合も，どこかで変数の値がおかしくなっている（偶然エラーの出ない程度の）ことが多いようです。

エラーを出した変数，おかしな動きをする原因の変数をつきとめて，その変数が出てくる行を全部確かめてください。Ver.3では，プログラム中の文字列を探す命令があるので便利です。

```
FIND "A"
```

とすると，プログラム中で「A」という文字がある行を全部表示してくれます。

このように，プログラムの誤りや入力ミスを「バグ」と呼び，それを訂正することを「デバッグ」と呼びます。

ファミコン緊急情報

わが地球防衛軍スタッフの連日にわたるファミコンとのたたかいで，以下のことが判明した。

①Ver.2でも後期に印刷されたマニュアルには書かれているのだが，「CALL」という命令がかくされていた。これはマシン語プログラムを動かす命令である。使い方は，ジャンプ編でくわしくのべよう。

②他にも「SPC」と「TAB」という命令が登録されていた。ただし，これは名前が登録されているだけで使えない。Ver.4以降にでも使うつもりかもしれない。

バグ情報－1

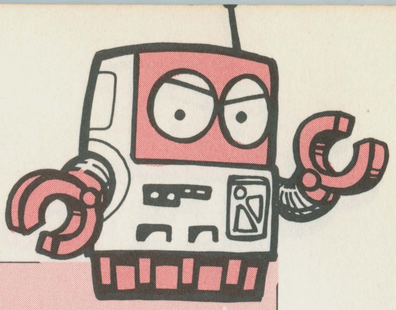
Ver.3で作ったプログラムをテープへSAVEしたものをVer.2でLOADすると，プログラムがメチャメチャになってしまう。というのは，ホップ編やソフト集001でもすでに説明したが，やっと原因がわかったので，説明しよう。

コンピューターが数字をメモリーへ入れる時，たとえば，

LOCATE Ø，1

などの「Ø」や「1」は，まず最初の場所に「次は10進法の数字だよ」というしるし(コード）を書いて，そのあと2バイトを使ってその数字を入れている。(つまり1つの数字を記憶するのに3バイト必要だ）ところが，Ver.3では，何を考えたか，1ケタの数字を記憶する時に限って，1バイトしか使わないように変更してしまったのだ。

たしかにメモリー数の少ないファミコンでは，メモリーの節約は非常にありがたいけど，この方式で書かれたテープをVer.2で読むと大変なことになる。Ver.2では，数字を表わすのに必ず3バイト使っているから，プログラムを見に行く順番がくるってし



まうのだ。

命令　すうじ　：　命令　　Ver.2

命令　1ケタの数字　：　命令　　Ver.3

Ver.3を持っている人は試してほしい。

```
10 LOCATE 10，12
10 LOCATE Ø，2
```

この2つのプログラムで残りバイト数（PRINT FREで出力する）が変わることがわかるだろう。

対策としては，

```
10 LOCATE Ø，1
```

のような1ケタの数字の前に，必ず

```
10 LOCATE &HØ，&H1
```

のように「&H」という文字を入れておけば大丈夫だ。

この「&H」というのは，16進数という数のかぞえ方を意味するやっかいなものだ（ジャンプ編でくわしく説明する)。ただし，どんなことがあっても，2ケタ以上の数字の前にはつけないこと（1Ø→&H1Øのように)。数字の中身が変わってしまうからだ。

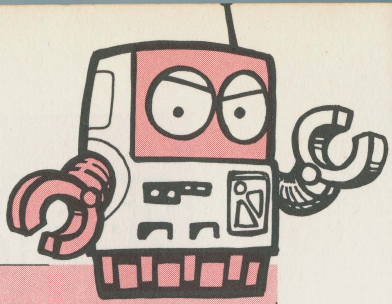
```
10 LOCATE 10，12
10 LOCATE &HØ，&H2
```

この2つでは，残りバイトが一緒だというのは調べればすぐわかる。

バグ情報－2

以下のプログラムを実行してみよう。

```
10 CLS
20 LOCATE 8，8
```



```
30 PRINT "TEST1"
40 LOCATE 8,9
50 COLOR 8,9,3
60 PRINT "TEST2"
```

「LOCATE」文で，X座標を「8」に指定しているにもかかわらず，60行のＰＲＩＮＴ文の文字列は，ずれて表示されている。

つまり，COLOR命令を実行するとカーソルの位置がずれてしまうのだ。

```
10 LOCATE 8,8
20 A=POS(0)
30 COLOR 8,8,3
40 B=POS(0)
50 CLS
60 PRINT A,B
```

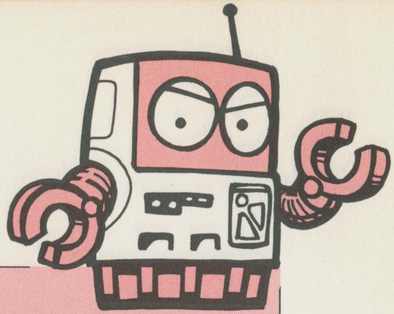
このプログラムでわかるだろう。このバグはVer.3では解消されているが，Ver.2ではLOCATE命令をともなったCOLOR文は必ず，PRINT文のうしろに置くことが必要だ。

バグ情報－3

```
10 READ A$
20 PRINT A$
30 DATA ャ
```

として「RUN」してみよう。ちゃんと「ャ」と表示されるはずだ。ここで「LIST」をとってみると，

```
10 READ A$
```



```
20 PRINT A$
30 DATA READ
```

となっている。

これは,「LIST」の処理ルーチンにバグがあるため,「DATA」のうしろにカタカナがあると，ものによっては中間言語（ファミリーベーシック・ジャンプ編を見て！）と勘違いしてしまうせいだ。

```
30 DATA "ャ"
```

のようにすれば問題はないが，普通ほとんどのパソコンも,DATA文の文字列は「“ ”」で囲まれていなくてもよいという約束なので困ってしまう。「ャ」以外にもこういう変化をするカタカナはいくつもある。探してみよう。また，変化する時としない時がある。なお，このバグはVer.3 でもとれていない。

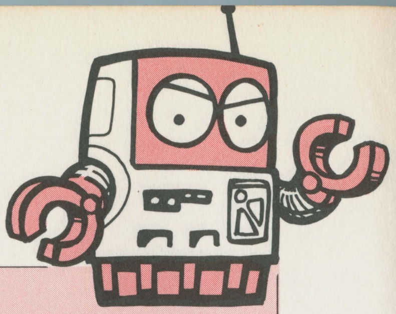
バグ情報ー4

どのコンピューターも,「英数字モード」と「カナ文字モード」のどちらの状態であっても，RETURNキーの機能は同じのはずだが，我々スタッフがEDIT（行の中身を修正する）中に,「カナ文字モードの状態でRETURNキーを押したあと「LIST」を取ると，修正されてない（RETURNキーの機能が働いてない）という現象が何回かあった。あとでのべる「熱暴走」のせいかもしれないが，重要な修正は「英数字モード」にしたあとRETURNキーを押した方が安心だ。

バグ情報ー5

次のプログラムを入力してほしい。

```
100 READ A$, B
110 PRINT A$, B
```



　200　DATA　"C"
　210　DATA　1

こうすると「A$」にはちゃんと「C」という文字が入っているけれど，数値変数「B」にはDATA文にある「1」ではなくて「0」が入っている。ところが，この200行と210行を

　200　DATA　"C",1

または，

　200　DATA　C
　210　DATA　1

のようにすると，「B」にはちゃんと「1」が入っている。

これは，おそらくDATA文の中で「" "」のあとに何もない時は，「，」の役割りも一緒にしているせいだと思われる。

つまり，

　200　DATA　"C",

と書いたのと同じわけだ。

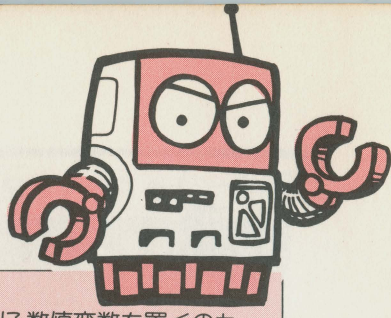
コンピューターは「C」という文字を読んだ（READ　A$）あと，その「C」のうしろに「，」があると思う。ところが「，」のうしろには何もないので，とりあえず「0」を次のREADに代入してしまうというわけだ。

おもしろいのは「" "」のうしろに「，」があって，他のデータが続くときは「" "」は「，」の働きをしなくなっている。

このバグを防ぐには，

　200　DATA　"C",1

のように「" "」のうしろに他のデータをつけるか，「" "」を使わないか，しかないが，バグ情報−3に書いたように，「" "」をつけないとカタカナの文字列がLISTをとる



とバケてしまう。DATAをREADする時，文字変数よりあとに数値変数を置くのも，プログラムの内容によっては不可能な時もある（RESTOREなどの時）。

「ソフト集002」では，ダミーのデータを1個おいているが，困ったバグだ。

このバグは，Ver.3ではなくなっているようだ。

バグ情報－6

これは，「バグ」ではないが，それ以上に

要　注　意　！

「熱暴走」という言葉を知っているだろうか。古いパソコンではよくあった。長時間コンピューターの電源を入れているうちに，コンピューターが熱を持ち，その熱がコンピューターに影響を与え，正常な動作をしなくなることがある。

現在，わがスタッフは12台のファミコンを持ち，日夜開発とデバッグにはげんでいるのだが，その内の約半数が時々暴走してしまうのだ。

プログラムを入力中に，突然，あの恐怖のオープニング画面と共に「ピコピコ」と音が出てきたり，カーソルが消えてキーボードをたたいても何の反応もなくなったりしてしまうのだ。

たしかに，徹夜を続けて何日間も電源を入れっぱなしの，かなりハードな使い方をしているのは事実だが，これではあまりにもスタッフがかわいそう。

電源はこまめに切りましょう。また，コンセントにさし込んだ電源アダプターも，ファミコンを使わない時はぬいておきましょう。

それにしても，少々故障率が高すぎる気もするのだが……。

キャラクタテーブルA



これらのキャラクタは，主にアニメキャラクタとして使用します。
各キャラクタの四隅に表記している数字は，DEF SPRITE文のCHR$（n）で使用する10進数です。

キャラクタテーブルB



検印省略　　NDC 548

ファミコン　ソフト集 002

昭和60年6月24日　　発　行

定価　800円

著　者　地球防衛軍

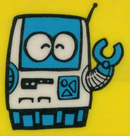
発行者　小川茂男

発行所　㈱誠文堂新光社
東京都千代田区神田錦町1－5－5
郵便番号 101
電話　東京　（292）　1211
振替口座　東京　7－6294
印刷・広研印刷株式会社
製本・岡嶋製本工業

ISBN4-416-18528-6 C2055

ファミコン大作戦

ファミコンソフト集 002



ファミコン大作戦

誠文堂新光社

ISBN4-416-18528-6 C-2055 ¥800E 定価 800 円

ファミコン大作戦

ファミコンソフト集 002

誠文堂新
社